CACiC 2023

CONGRESO ARGENTINO
DE CIENCIAS
DE LA COMPUTACIÓN

9 al 12 de Octubre de 2023

Universidad Nacional de Luján

Luján, Argentina

2023

RedUNCI

Licenciatura en Sistemas de Información

DEPARTAMENTO CIENCIAS BASICAS

UNIVERSIDAD NACIONAL DE LUJÁN

cacic2023.unlu.edu.ar

XXIX Congreso Argentino de Ciencias de la Computación:

Libro de Actas

ISBN 978-987-9285-51-0

Fernández, Juan Manuel
Libro de Actas: XXIX Congreso Argentino de Ciencias de la Computación - CACIC 2023 / Juan Manuel Fernández ; Compilación de Juan Manuel Fernández. - 1a ed. - Luján : Universidad Nacional de Luján, 2024.
Libro digital, PDF

Archivo Digital: descarga y online
ISBN 978-987-9285-51-0

1. Computación. I. Fernández, Juan Manuel, comp. II. Título.
CDD 004.25



El **XXIX Congreso Argentino de Ciencias de la Computación** (**CACiC 2023**) se llevó a cabo, por primera vez en la *Universidad Nacional de Luján*, entre los días ***9 y 12 de octubre de 2023***, en la ciudad de *Luján*, Provincia de Buenos Aires, ***Argentina***.

El evento Nacional de *Ciencias de la Computación* de mayor concurrencia y de alcance federal fue punto de encuentro de investigadores, docentes, estudiantes de grado/postgrado y profesionales en las Ciencias de la Información.

Organizado por la ***Red de Universidades Nacionales con carreras en Informática*** (***RedUNCI***) cubrió diferentes áreas a través de Workshops, coordinados por expertos en los temas del área, donde se presentaron trabajos científicos evaluados por investigadores del país y del exterior que se compilan en el presente Libro de Actas.

Organizaron:

RedUNCI

Licenciatura en Sistemas de Información

DEPARTAMENTO CIENCIAS BASICAS

UNIVERSIDAD NACIONAL DE LUJÁN

https://cacic2023.unlu.edu.ar/

Comité Académico

- UBA - FCEyN | Ceria, Santiago
- UBA - FI | Echeverria, Adriana
- UNLP | Pesado, Patricia
- UNS | Rueda, Sonia
- UNSL | Printista, Marcela
- UNNOBA | Aciti, Claudio
- UNComa | Grosso, Guillermo
- UNLaM | Eterovic, Jorge
- UNLPam | Alfonso, Hugo
- UNTDF | Koremblit, Gabriel
- UNSa | Gil, Gustavo
- UNPA | Lasso, Marta
- UNSJ | Rodriguez, Nelson
- UADER | Mengarelli, José Luis
- UNP | Buckle, Carlos
- UNER | Tugnarelli, Mónica
- UNNE | Dapozo, Gladys
- UNR | Casali, Ana
- UNAM | Caballero, Sergio
- UNNOBA | Russo, Claudia
- UNdeC | Carmona, Fernanda
- UNLa | Azcurra, Diego
- UNSE | Figueroa, Liliana
- UNDEF | Cipriano, Marcelo
- UNL | Loyarte, Horacio
- UNRC | Arroyo, Marcelo
- UNC | Fridlender, Daniel
- UNJu | Herrera Cognetta, Analía
- UNRN | Vivas, Luis
- UNVM | Prato, Laura
- UNLu | Fernandez, Juan Manuel
- UNCA | Poliche Maria Valeria
- UNLaR | Molina, Miguel
- UNTREF | Oliveros, Alejandro
- UNT | Luccioni, Griselda María
- UNAJ | Morales, Martín
- UNCAUS | Zachman Patricia
- UNO | Foti, Antonio
- UNCUYO | Haderne, Marisa Fabiana
- UNMDP | Ríos, Carlos
- UNQ |
- UNAHUR | Puricelli, Fernando
- UNSAdA | Ramón, Hugo
- UNSAM | Estayno, Marcelo
- UMET | Quiroga, Flavia
- UNDAV | Mayer, Roberto
- UM | Chapperon, Gabriela
- UAI | De Vincenzi, Marcelo
- UB | Guerci, Alberto
- UK | Panizzi, Marisa
- UAP | Bournissen Juan
- CAECE | Malbernat, Lucía
- UP | Alvarez Adriana
- UCA Rosario | Grieco, Sebastián
- USAL | Zanitti, Marcelo
- UdA | Giménez, Rosa
- UGD | Ruidías, Hector Javier
- UADE | Feijó, Daniel
- UCEMA | Guglianone, Ariadna
- UA | Cosentino, Juan Pablo
- UAA | Rathmann, Liliana
- UCA | Bertone, Rodolfo
- ITBA | Bolo, Mario
- UCH | Brachetta Mariana
- UM | Asensio, Mariela

Comité Científico

- Abásolo, Ma. José (*Argentina*)
- Aciti, Claudio (*Argentina*)
- Alfonso, Hugo (*Argentina*)
- Ardenghi, Jorge (*Argentina*)
- Arroyo, Marcelo (*Argentina*)
- Astudillo Hernán (*Chile*)
- Baldasarri, Sandra (*España*)
- Balladini, Javier (*Argentina*)
- Barbosa, Luis (*Portugal*)
- Bertone, Rodolfo (*Argentina*)
- Bría, Oscar (*Argentina*)
- Brisaboa, Nieves (*España*)
- Buckle, Carlos (*Argentina*)
- Cañas, Alberto (*EEUU*)
- Casali, Ana (*Argentina*)
- Castro, Silvia (*Argentina*)
- Cechich, Alejandra (*Argentina*)
- Chavez, Edgar (*México*)
- Coello Coello, Carlos (*México*)
- Cuckierman, Uriel (*Argentina*)
- De Giusti, Armando (*Argentina*)
- De Giusti, Laura (*Argentina*)
- De Vincenzi, Marcelo (*Argentina*)
- Deco, Claudia (*Argentina*)
- Depetris, Beatriz (*Argentina*)
- Diaz, Javier (*Argentina*)
- Dix, Juerguen (*Alemania*)
- Doallo, Ramón (*España*)
- Docampo, Domingo (*España*)
- Dujmovic Jozo (*EEUU*)
- Estayno, Marcelo (*Argentina*)
- Estevez, Elsa (*Argentina*)
- Eterovic, Jorge (*Argentina*)
- Falappa, Marcelo (*Argentina*)
- Fillottrani, Pablo (*Argentina*)
- Finochietto, Jorge (*Argentina*)
- Frati Emmanuel (*Argentina*)
- Fridlender Daniel (*Argentina*)
- García Garino, Carlos (*Argentina*)
- García Villalba, Javier (*España*)
- Género, Marcela (*España*)
- Gomez, Sergio (*Argentina*)
- Gröller, Eduard (*Austria*)
- Guerrero, Roberto (*Argentina*)
- Ierache, Jorge (*Argentina*)
- Janowski, Tomasz (*Naciones Unidas*)
- Kuna Horacio (*Argentina*)
- Lanzarini, Laura (*Argentina*)
- Leguizamón, Guillermo (*Argentina*)
- Lopez Gil, Fernando (*España*)
- Loui, Ronald Prescott (*EEUU*)
- Luque, Emilio (*España*)
- Madoz, Cristina (*Argentina*)
- Malberti, Alejandra (*Argentina*)
- Manresa Yee, Cristina (*España*)
- Marín, Mauricio (*Chile*)
- Mas Sansó, Ramón (*España*)
- Micolini, Orlando (*Argentina*)
- Mon Alicia (*Argentina*)
- Motz, Regina (*Uruguay*)
- Naiouf, Marcelo (*Argentina*)
- Navarro Martín, Antonio (*España*)
- Olivas Varela, José Angel (*España*)
- Pardo, Álvaro (*Uruguay*)
- Pasini, Ariel (*Argentina*)
- Pesado, Patricia (*Argentina*)
- Piattini, Mario (*España*)
- Piccoli, María Fabiana (*Argentina*)
- Printista, Marcela (*Argentina*)
- Puppo, Enrico (*Italia*)
- Ramón, Hugo (*Argentina*)
- Rexachs, Dolores (*España*)
- Reyes, Nora (*Argentina*)

- Roig Vila, Rosabel (*España*)
- Rossi, Gustavo (*Argentina*)
- Rosso, Paolo (*España*)
- Rueda, Sonia (*Argentina*)
- Russo, Claudia (*Argentina*)
- Salto, Carolina (*Argentina*)
- Sanz, Cecilia (*Argentina*)
- Simari, Guillermo (*Argentina*)
- Steinmetz, Ralf (*Alemania*)
- Suppi, Remo (*España*)
- Tarouco, Liane (*Brasil*)
- Thomas, Pablo (*Argentina*)
- Tirado, Francisco (*España*)
- Velho, Luiz (*Brasil*)
- Vendrell, Eduardo (*España*)
- Vénere, Marcelo (*Argentina*)
- Villagarcía, Horacio (*Argentina*)
- Zanarini, Dante (*Argentina*)

Autoridades UNLu

Rector:
Lic. Walter Panessi

Vicerrectora:
Prof. María Esther Leguizamón

Decana de Ciencias Básicas:
Lic. Emma Ferrero

Vicedirector Decano de Ciencias Básicas:
Dr. Javier Di Salvo

Secretario Académico de Ciencias Básicas y Coordinador de la Licenciatura en Sistemas de Información:
Mg. Juan M. Fernández

Secretario Académico de Rectorado:
Bioq. Jorge Mufato

Secretaría de Posgrado:
Lic. Hugo Delfino

Secretaría de Ciencia y Tecnología:
Dr. Juan Carlos Fernícola

Subsecretaría de Gestión Académica:
Lic. Carla Martínez

Subsecretaría de Planificación Académica:
Prof. Estela Salles

Secretaría de Administración:
Lic. Alicia Segura

Secretaría de Bienestar Universitario y Asuntos Estudiantiles:
Lic. Luciano Quarenta

Secretaría de Extensión e Integración con la Comunidad:
Dra. Florencia Cendali

Secretaría de Cooperación Internacional e Internacionalización:
Mg. Pablo Albertti

Comité Organizador

- Juan M. Fernández
- Claudia Ortiz
- Pablo Cesar Chale
- Gabriel Tolosa
- Fernando Bordignon
- Fabian Borea
- Isabel Del Buono
- Rosa Cicala
- Silvina Casablanca
- José Luis Caero
- José Racker
- Joaquín Belgrano
- Jorge Mufato
- Carla Martinez
- Florencia Cendali
- Daniel Lanson
- Guillermo Cherencio
- Gladys Kaplan
- Rosana Matuk
- María Rosana Mason
- Pablo Tomás Delvechio

Autoridades Red UNCI

Coordinador titular:

Pesado, Patricia (UNLP) [2022-2024]

Coordinador Alterno:

Eterovic, Jorge (UNLZ) [2022-2024]

Junta Directiva

- **Estayno Marcelo** (UNSAM) [2022-2024]
- **Printista, Marcela** (UNSL) [2022-2024]
- **Tugnarelli, Mónica** (UNER) [2022-2024]
- **Malbernat, Lucia** (CAECE) [2022-2024]
- **Arroyo, Marcelo** (UNRC) [2022-2023]
- **Aciti, Claudio** (UNICEN) [2021-2023]
- **Lasso, Marta** (UNPA) [2021-2023]
- **Panizzi, Marisa** (UK) [2021-2023]

Miembro Honorario

De Giusti, Armando (UNLP)

Secretarías

Secretaría Administrativa:

Carmona Fernanda

Secretaría Académica:

Russo, Claudia

Secretaría de Ciencia y Técnica:

Rodríguez, Nelson

Secretaría de Asuntos Reglamentarios:

De Vincenzi, Marcelo

Secretaría de Vinculación Tecnológica y Profesional:

Marcelo Estayno

Secretaría de Congresos, Publicaciones y Difusión:

Thomas, Pablo

Índice

XVIII Workshop de Arquitecturas, Redes y Sistemas Operativos > Full Papers 1

Caso de estudio: metodología para el diseño y desarrollo de sistemas embebidos distribuidos 2
Ing. Luis Orlando Ventre, Dr. Ing. Orlando Micolini, Ing. Mauricio Ludemann, Agustín Carranza, David D'Andrea, Enzo Candotti

XVIII Workshop de Arquitecturas, Redes y Sistemas Operativos > Short Papers 10

CRANE: Simplificando el Despliegue de Aplicaciones Contenerizadas en Entornos Locales 11
Jose Miguel Silva Pavón, Franco Bellino, Patricia Bazán, Alejandra B. Lliteras, Nicolás del Rio

XXIV Workshop de Agentes y Sistemas Inteligentes > Full Papers 17

Generación de gestos de lengua de señas con redes neuronales generativas basadas en poses y etiquetas 18
Gaston Gustavo Rios, Pedro Dal Bianco, Franco Ronchetti, Facundo Quiroga, Oscar Stanchi, and Waldo Hasperué

Mejorando la Identificación de Marcas de Ganado Vacuno: Redes Siamesas en el Aprendizaje de Funciones de Distancia 27
Federico Stauber, Adrián Planas, Andrés Pascal

Ciencia de Datos para el Desarrollo de un Modelo Predictivo de Heladas 38
María Isabel Masanet, Raúl Orcar Klenzi

On the Precision Evaluation in Non-linear Sensor Network Design 48
José Hernández, Silvia Simón, Mercedes Carnero,Gabriela Minetti, Carolina Salto

Variantes Dispersivas para el Operador de Mutación en Algoritmos Genéticos con Cromosmas Binarios 58
Ana Laura Maffei y Tomás Tetzlaff

XXIV Workshop de Agentes y Sistemas Inteligentes > Short Papers 66

Utilizando Argumentación Rebatible en la Detección y Respuesta ante Intrusión en Sistemas Biométricos 67
Graciela R. Etchart, Juan C.L. Teze, Carlos E. Alvez, María V. Martinez, Gerardo I. Simari

Adapting PALS to solve VRP with Time Windows 72
C. Bermudez, H. Alfonso, G. Minetti, and C. Salto

Aprendizaje no supervisado combinando autoencoders y agrupamiento 77
Lucas Cochella, Laura Lanzarini and Patricia Jimbo Santana

Analysis of Bioinformatic algorithms for MSA 81
Adrián Díaz and Gabriela Minetti

XXI Workshop de Bases de Datos y Minería de Datos > Full Papers 87

Descubrimiento de patrones relevantes del Dengue mediante la Programación Lógica Inductiva y Twitter 88
Carlos Alberto Benítez Galván, Tania Mabel Leguizamón Ovelar, Gustavo Sosa-Cabrera, and María E. García-Díaz

Performance analysis of the Survival-SVM classifier applied to gene-expression databases 97
Genaro Camele and Waldo Hasperué

Replicación sincrónica y asincrónica de Bases de Datos en Modalidad "Maestro-Esclavo", caso de estudio: Firebird SQL Server 106
Guillermo Ruben Cherencio, Mario Gerardo Perello, Juan Carlos Romero

On the Assessment of Personality Traits by Using Text Mining Techniques 114
Luis Montenegro, Maximiliano Sapino, Edgardo Ferretti, and Leticia Cagnina

Impacto de Factores Topológicos y de Desbalance en la Clasificación de Nodos con GCNs 124
Tatiana S. Parlanti, Carlos A. Catania, and Luis G. Moyano

Aplicación de Procesos de Diseño de Bases de Datos NoSQL 134
Luciano Marrero, Verena Olsowy, Fernando Tesone, Pablo Thomas, Patricia Pesado

Primeras Experiencias en la Identificación de Personas con Riesgo de Diabetes en la Población Argentina utilizando Técnicas de Aprendizaje Automático 146
Enzo Rucci, Gonzalo Tittarelli, Franco Ronchetti, Jorge F. Elgart, Laura Lanzarini and Juan José Gagliardino

XXI Workshop de Computación Gráfica, Imágenes y Visualización > Full Papers **159**

Visual Analysis of Temporal Data Associated with Cryptocurrencies 160
Mercedes Barrionuevo and María Luján Ganuza

Detección de daños con visión artificial en inspecciones marítimas: un mapeo sistemático de la literatura 170
Hugo Daniel Flores, Carlos Neil

XII Workshop de Innovación en Educación en Informática > Full Papers **181**

Estilos de aprendizajes personalizados con inteligencia artificial. Un mapeo sistemático de la literatura 182
Fabián Maffei, Carlos Neil, Nicolás Battaglia

Desarrollo de competencias conducido por rúbricas 195
Carlos Neil, Nicolás Battaglia, Marcelo De Vincenzi

Desarrollo de Sistemas Inteligentes: experiencias en capacitación 205
Analía Amandi, Nelson Acosta, Marcelo Campo

Hacia una Educación Integral: Modelo Visual para la Interconexión de Conceptos 217
Lucas Videla, Verónica Inés Aubin and Renata Silvia Guatelli

Aprendizaje Basado en Proyectos como propuesta de enseñanza para una asignatura introductoria de programación 227
Victoria Aragón and Gisela Dorzán

Enseñanza del Diseño de Aplicaciones Web mediante el Proceso Iconix Adaptado el Framework Odoo 236
Alejandro Sartorio, Matias Banega

Programación por pares como estrategia de enseñanza en contextos de masividad 248
Gladys Dapozo, Cristina Greiner, Raquel Petris, Ana María Company, María Cecilia Espíndola

XII Workshop de Innovación en Educación en Informática > Short Papers **257**

Propuesta para la Enseñanza de Desarrollo de Software Guiado por Pruebas 258
Nicolás Paez, Alejandra Zangara, Diego Fontevila, and Alejandro Oliveros

XX Workshop de Ingeniería de Software > Full Papers **263**

Conflict resolution for aspect-oriented static modeling 264
Fernando Pinciroli, Laura Zeligueta and Marcelo Palma

Addressing the State Explosion Problem for Big Data Systems Formal Verification 279
Fernando Asteasuain

Gestión de los requerimientos no funcionales en equipos ágiles: un caso de estudio..... 289
Lourdes Romera, Jeremías González, Andrea Lezcano, Juan A. Carruthers, Emanuel Irrazábal

Estudios de casos realizados en PyMEs de Argentina para validar un conjunto de métricas para el despliegue de sistemas de software..... 299
Pablo Vázquez, Marisa Panizzi, Rodolfo Bertone

Identificación Top-Down de Variedad de Contexto: Un Caso de Estudio en Fluctuaciones de la Napa Freática.....308
Mariano Campetella, Alejandra Cechich, Agustina Buccella, Ayelén Montenegro, Ángel Muñoz and Andrea Rodríguez

Análisis comparativo de alternativas para la detección y comunicación con dispositivos BLE Beacons en Aplicaciones Móviles Multiplataforma..... 318
Juan Fernández Sosa, Santiago Medina, Pablo Thomas, Leonardo Corbalán, Marcelo Naiouf

An empirical study of API Management and ISO/IEC SQuaRE: a practitioners' perspective 332
Eder dos Santos and Sandra Casas

Evaluación de requisitos de software utilizando GQM..... 342
Gladys Kaplan, Gerardo Riera, María Inés Bertín, Gabriel Blanco

Estudio de Técnicas de Procesamiento de Lenguaje Natural en la Validación de Requerimientos.....352
Sonia Santana, Lucrecia Perero, Noelia Rodriguez, Leandro Antonelli,Alejandro Fernandez

Calidad 4.0 en Ingeniería de Software: propuesta de un marco de trabajo 365
Kristian Petkoff Bankoff, Rocío Muñoz, Ariel Pasini, and Patricia Pesado

XX Workshop de Ingeniería de Software > Short Papers **376**

Refinamiento del método de análisis de factibilidad del despliegue de sistemas de software: Estudio de caso.....377
Leandro Moreno, Marisa Panizzi , Rodolfo Bertone

Encuesta sobre el uso y la relación entre Escenarios y User Story Mapping..... 382
Andrea Alegretti, Leandro Antonelli, Marisa Panizzi

XVI Workshop de Innovación en Sistemas de Software > Full Papers..... **387**

Bot de Trading. Control movimientos de Cripto Monedas. 388
Juan Pablo Pollini, Facundo Ferro, Marcelo Taruschio, Rodolfo Bertone

Diseñar software accesible: pensar fuera de la caja 398
Claudia Ortiz, Cecilia Challiol, Walter Panessi

A Prototypical Tool for Analyzing Functional Dependencies Induced from Spreadsheets.....409
Sergio Alejandro Gómez and Pablo Rubén Fillottrani

Proyecto Aplicación para la Gestión de Residuos Sólidos Urbanos..... 418
Lic. Diego Martín Luna, Ing. Franco Pablo Quintero, Mg. Rodolfo Bertone, C.C. Marcelo Taruschio

Digitalización 3D para aplicaciones móviles con realidad virtual y realidad aumentada 428
Agustín Cao, Sebastián Dapoto, Pablo Thomas, Patricia Pesado

An integration of Brain-Computer Interfaces with an Alternative Augmentative Communication System 439
Javier F. Díaz, Laura A. Fava, Ivana Harari, Fernando Martínez, Miguel Tellechea

XVI Workshop de Innovación en Sistemas de Software > Short Papers.....450

Itinerario: una Apps de Nueva Generación..... 451
Adriana Elizabeth Martín; Susana Beatriz Chavez; Sergio Rafael Flores; A. Sara Zogbe

XXIV Workshop de Procesamiento Distribuido y Paralelo > Full Papers.....456

Evaluación de la inicialización y el arranque en frio de los lenguajes de programación en una plataforma serverless. Amazon Web Services como caso de estudio..... 457
Matías Rodríguez, Nelson Rodríguez, María Murazzo

Detección de Vulnerabilidades en Smart Contracts Usando Machine Learning a Nivel de Bytecode..... 467
Matias A. Carballo, Hernan D. Merlino

Implementando estrategias de resiliencia en una arquitectura basada en microservicios..... 476
Sergio Leonel Suárez, Enzo Rucci, Victor Betran and Diego Montezanti

Adaptación de Algoritmo OpenMP para Computar Caminos Mínimos en Grafos en Arquitecturas x86..... 489
Sergio Calderón, Enzo Rucci, and Franco Chichizola

Evaluation of FaaS as an alternative to build HPC environments..... 501
María Murazzo, Joaquín Lebetti, Nelson Rodríguez, Adriana Martin

XIV Workshop de Procesamiento de Señales y Sistemas de Tiempo Real > Full Papers513

ADS-B Collaboration with SSR..... 514
Oscar Bria and Javier Giacomantone

Determinación de la Orientación en Vehículos No Tripulados..... 524
Ignacio Zaradnik, Luciano Salvatore, Oreste Daniel Lupi, Diego Turconi, Juan Pablo Calabro

Experiencia de desarrollo de una aplicación de reconocimiento de canciones mediante la técnica de huella de audio.....531
Rodrigo Lago, Veronica Scholz, Roman Bond, Martin Morales, Diego Encinas

Sistema de Adquisición de Datos Utilizando Sensores de Baja Complejidad para Tractores Antiguos del Alto Valle de Río Negro y Neuquén..... 541
Rafael Ignacio Zurita, Naiara Sheffield, Marcelo Moreyra, Dario Mendieta, Favio Masson, and Miriam Lechner

XII Workshop de Seguridad Informática > Full Papers.....551

Estudio exploratorio sobre la adopción de la Ciberseguridad en PyMES del partido de Hurlingham.....552
Marisa Panizzi, Felipe Ortiz, Iris Sattolo, Fernando Puricelli, Rodolfo Bertone

Cifrador de Bloque con Alternancia de Matrices de Permutación y Combinación de Algoritmos Cifradores Paralelos..... 562
Andrés Francisco Farías, Andrés Alejandro Farías

Simulación de Montecarlo, con Dos Variables Aleatorias, Utilizando Generador Binario Pseudoaleatorio, para Problemas de Reparación de Máquinas.....572
Andrés Francisco Farías, Germán Antonio Montejano, Ana Gabriela Garis[3], Pablo Marcelo García, Andrés Alejandro Farías

Generador Binario Pseudoaleatorio Basado en la Combinación de Registros de Desplazamiento de Retroalimentación Lineal con Función de Filtrado No Lineal 582
Andrés Francisco Farías, Andrés Alejandro Farías

Generación De Nombres De Dominios Mediante Redes Neuronales Híbridas CNN-LSTM..... 591
Reynier Leyva La O, Rodrigo Gonzalez, and Carlos A. Catania

XXII Workshop de Tecnología Informática Aplicada en Educación > Full Papers..... 603

Laboratorio virtual de electrónica básica para alumnos universitarios dentro de aula extendida..... 604
Mg. Leonardo Navarria, Mg. Alejandro González, Dra. Alejandra Zangara

Propuesta STEAM para la formación científica de estudiantes pertenecientes a carreras del área de las ciencias exactas y naturales..... 615
Silvina Manganelli

Incorporación a un serious games de un proceso evaluativo basado en analíticas de aprendizaje y lógica difusa..... 623
Leonel Guccione, Stella Massa, Guillermo Lazzurri, Adolfo Spinelli, Franco Kühn

Herramienta Práctica de Aplicación de Emparrillado en la Asignatura Inteligencia Artificial..... 631
Federico Ilan Maleh, Cinthia Vegega, Pablo Pytel, María F. Pollo-Cattaneo

eUCCvm: Una Herramienta Educativa Integral para la Programación de Sistemas Embebidos..... 641
Cristian F. Perez-Monte, Gustavo Mercado, Carlos Taffernaberry, Ana Laura Diedrichs, Fabiana Piccoli, Mario Sebastian Tobar, Marcelo Ledda, Raúl Moralejo and Rodrigo Gonzalez

Metodologías de evaluación de la participación en foros de debate académicos: Una revisión sistemática..... 652
Paula Dieser, Cecilia Sanz, Alejandra Zangara

Generación de STEAM en condiciones extremas con xSTEAM..... 662
Gilda R. Romero, Cecilia Salina, Martín Domen, Daniela López De Luise, Mauro Bejarano

Aprendizaje y Enseñanza de Programación: El desafío de herramientas de Inteligencia Artificial como ChatGPT..... 672
Laura De Giusti, Gonzalo Lujan Villarreal, Eduardo Ibañez, Armando De Giusti

VIST: Herramienta para la Visualización del Seguimiento de la Interacción en Trabajo colaborativo en Moodle..... 684
Mitchell Vásquez-Bermúdez, Cecilia Sanz, María Alejandra Zangara, Jorge Hidalgo

Aplicaciones educativas y realidad aumentada: Un mapeo sistemático de literatura..... 694
Paulina Giménez-Suárez, Rosanna Costaguta y María de los Ángeles Menini

Una revisión sobre enfoques de agrupamiento automático en base a la personalidad y los estilos de aprendizaje de los estudiantes..... 701
César Ponce-Padilla, Germán Lescano y Rosanna Costaguta

Uso de videojuegos en contextos de aprendizaje colaborativo: Un mapeo sistemático de literatura..... 711
Guillermo González, Walter Martin Anriquez-Atia y Rosanna Costaguta

EVOTEM: una herramienta de software para el análisis de la evolución temática de publicaciones científicas en español..... 720
Santiago Bianco, Laura Lanzarini, and Alejandra Zangara

INTEGRA+506: Una Metodología para el diseño y desarrollo de juegos serios 730
Juan Carlos Sandí-Delgado and Patricia Bazán

XXII Workshop de Tecnología Informática Aplicada en Educación > Short Papers..... 744

Diseño de sistemas recomendadores en el ámbito de las bibliotecas universitarias 745
Sergio A. Cabrera, Cristian D. Pacífico y Juan C. L. Teze

VI Track de Gobierno Digital y Ciudades Inteligentes > Full Papers..... 751

Asistencia posicional para personas ciegas o con baja visión en edificios públicos *752*
Federico Cristina, Sebastián Dapoto, Ramiro Intas

La gobernanza en los Ecosistemas Digitales de Integrabilidad a la luz de la innovación tecnológica..... *764*
Giorgetti Gustavo, Luz Clara Horacio , Bazán Patricia

Contribución del proyecto CAP4CITY a los ODS 2030 ..... *774*
Rocío Muñoz, Ariel Pasini, Patricia Pesado

Track de Short Papers Alumnos..... 784

Hacia el aprendizaje de la Revolución de Mayo en nivel primario mediante un Chatbot de Lenguaje Natural..... *785*
Sergio Orciari, Marisa Panizzi, Iris Sattolo

Sistema IoT para Control y Mantenimiento de Cultivos Hidropónicos NFT *790*
Ariana Mazzini, Facundo Ortega, Matias Solana Mendez, Melisa Kuzman

BreatheEasy: Sistema de monitoreo ambiental distribuido para la detección y alerta temprana de fugas de gases tóxicos..... *797*
Franco Bandieri, Nicolás Falcone, Melisa Kuzman

Tecnología colaborativa para investigadores: implementación y optimización de un repositorio GitLab para gestión de la configuración y versionado *803*
Damián Ferrara, Emanuel Rodriguez, Romeo Lorenzo Monfroglio, Manuel Robles, María Ayelén Díaz Lapérgola, Leopoldo Nahuel

Propuesta de Evaluación de Experiencia de Usuario Usando Detección Facial de Emociones..... *808*
Sergio E. Ledesma Vila, Laura N. Aballay, Flavia Millan

I+IoT RLab Laboratorio Remoto de IoT e IIoT, para la enseñanza, la asistencia y consultoría a empresas..... *814*
Gabriel Ambrosio, Mariano Zapata, Gustavo Mercado, Carlos Taffernaberry

Reconocimiento de Logotipos de Empresas Tecnológicas mediante Redes Neuronales Artificiales *820*
Ignacio Ibañez, Juan Bautista Rodriguez, Cristian Desiderio Arana, CinthiaVegega, María F. Pollo-Cattaneo

Análisis Discriminante sobre Datos de Egresados Universitarios para la Toma de Decisiones Institucionales..... *826*
Simón Pedro González, Myriam Herrera y Susana Ruiz

Desarrollo de un chatbot y virtualización basada en contenedores para una aplicación web de gestión de laboratorios *833*
María del Valle Scheffer, Ilda Flavia Millán, Fabio Gabriel Sanmartino

Comparación de modelos de detección de objetos aplicados a una plantación de kiwi *838*
Dante Mendoza, Camila Pagano, Silvia Pérez

EcoConversa: Contribuyendo al futuro sostenible con Inteligencia Artificial.... *843*
Franco Brandan; Andrea Villagra y Daniel Pandolfi

Sistema de control y monitoreo para Biorreactor con tecnología IoT *848*
Mauro Escalante, Diego Irusta, Gustavo Mercado, Carlos Taffernaberry, Ana Laura Diedrichs, Sebastian Tobar, Cristian Pérez Monte, Eugenio Fisicaro, Adrian Carmona, Ana Lattuca, Nelson Merino, Néstor Manzur, Antonio Álvarez

Generación de documentación automática para sistemas embebidos utilizando uModelFactory..... *854*
Nicolas Tobías Almaraz

Plataforma web para el área de deportes electrónicos 859
Lorenzo Handula, Luciano Iglesias, Cecilia Sanz

Desarrollo de un modelo con XGBoost intérprete de la Lengua de Señas Argentina 866
Facundo Recabarren, Fabrizio Amaya, Raúl Klenzi, María Isabel Masanet

Modelos de Machine Learning para Vehículos Autónomos aplicados a la Detección de Obstáculos en Centros de Logística 871
Débora Denise Altamirano Medina, Augusto Guillermo Tapparelli, Matías Kreder y Pablo Ezequiel Inchausti

Índice de Autores **876**

Coordinadores:
- Carlos Buckle (UNPSJB)
- Marcelo Arroyo (UNRC)
- Jorge Ardenghi (UNS)

CACiC 2023

CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

XVIII Workshop de Arquitecturas, Redes y Sistemas Operativos

> Full Papers

WARSO

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Caso de estudio: metodología para el diseño y desarrollo de sistemas embebidos distribuidos

Ing. Luis Orlando Ventre[1], Dr. Ing. Orlando Micolini[1], Ing. Mauricio Ludemann[1], Agustín Carranza[1], David D'Andrea[1], Enzo Candotti[1]

[1] Laboratorio de Arquitectura de Computadoras,
FCEFyN-Universidad Nacional de Córdoba
Av. Vélez Sarsfield 1601, CP-5000, Córdoba, Argentina
{luis.ventre, orlando.micolini, mauri.ludemann}@unc.edu.ar

Abstract. En este proyecto se introduce una metodología que se aplica al diseñoy desarrollo de un sistema de control de acceso embebido y distribuido. Esta metodología facilita desacoplar la lógica, la política de resolución de conflictos y las acciones, lo que da lugar a un sistema modular, simple, mantenible, formaly flexible. Además, se logra la verificación formal de la lógica en todas las fasesdel desarrollo. Para modelar la lógica del sistema, se emplean redes de Petri y seconvierten en código ejecutable mediante la ecuación de estado generalizada. Esta solución consigue conservar las propiedades verificadas mediante el uso del formalismo matemático. La implementación incluye un monitor de concurrenciaque integra los diversos componentes de software y hardware del sistema. Asimismo, se establecen interfaces definidas entre los dispositivos e incorpora librerías y protocolos estándares. Además, se destacan los beneficios de aplicar la metodología propuesta al diseño de sistemas críticos y reactivos. Se evidencia su facilidad para abordar problemas complejos, garantizando la escalabilidad y la fiabilidad del sistema desarrollado.

Keywords: Metodología, diseño, sistemas embebidos, Redes de Petri, sistemasdistribuidos

1. Introducción

Un sistema distribuido es una colección de sistemas con capacidad de computó independientes interconectados en red que aparentan ser un único sistema coherente para sus usuarios. El objetivo principal es facilitar el acceso a recursos, y compartirlos de una manera eficiente y controlada. Son transparentes, ocultan los procesos y recursos físicamente distribuidos. Además, son *escalables*, respecto de su tamaño, su distribución geográfica y la capacidad de mantener su administración controlada [1].

En el desarrollo de sistemas embebidos y distribuidos se utilizan técnicas de co-diseño de hardware software y lenguajes de programación como C, C++, Java, VHDL y Verilog. Sin embargo, la codificación manual tiene desventajas dada la propensión a errores, la limitada mantenibilidad y escalabilidad, la alta volatilidad en los requerimientos y los elevados recursos necesarios para la validación y verificación a través de simulaciones y testing. En los sistemas embebidos complejos, se requiere de un proceso de codificación iterativo, para la corrección de errores o cambio en requerimientos, así como la validación y testing de prototipos; dicha metodología solo

es capaz de mitigar la ausencia de errores. Estas tareas impactan significativamente en el tiempo involucrado para el desarrollo del sistema [2].

Para el desarrollo de este trabajo se utiliza diseño basado en modelos [3]. Es un método visual y/o matemático para comprender y resolver diseños asociados con sistemas embebidos complicados. Esta metodología engloba otros conceptos importantes como "arquitectura dirigida por modelos" [4]. El diseño basado en modelos utiliza éstos para dar soporte a otras etapas del desarrollo como por ejemplo la simulación, validación, verificación e implementación.

Las redes de Petri (RdP) [5], son un formalismo de modelado que soporta el desarrollo de sistemas basado en modelos. Este formalismo gráfico-matemático soporta explícitamente y facilita el modelado de: concurrencia, conflictos, recursos compartidos, exclusión mutua y sincronización. Es importante notar que las RdP tienen claramente definida su semántica de ejecución, y su representación matemática, soportando rigurosa documentación, simulación, verificación y traducción a código de ejecución [6]. Además, las RdP son un formalismo matemático abstracto por lo que son intrínsecamente independientes de la plataforma logrando así flexibilidad para alcanzar diversos objetivos de performance, costos, consumo de energía entre otros.

En este informe se presenta un caso de estudio donde se usa una metodología, que posibilita la automatización de la codificación del control de una planta modelada con RdP no autónomas. Esta metodología evita los inconvenientes asociados a la codificación manual de la lógica, y reduce significativamente el tiempo requerido para las etapas de simulación y prototipos. Además, garantiza la validación y verificación matemática del modelo lógico, lo que implica que la especificación documenta fielmente la implementación real.

2. Desarrollo

Para el desarrollo de este sistema distribuido resulta fundamental definir interfaces e implementar protocolos estándares para los módulos de software. Esto, sumado al formalismo matemático de las RdP, colaboran en la obtención de un diseño compacto, robusto y seguro.

2.1 Arquitectura de la Solución

El principal objetivo de la metodología propuesta consiste en descomponer y desacoplar el sistema en: eventos o estímulos, estados, lógica, política y acciones; como se muestra en la Fig. 1.

El punto de partida del diseño es el bloque monitor de concurrencia etiquetado en la Fig. 1 como monitor, el cual será responsable de la gestión de los eventos en sección crítica, sincronización y resolución de conflictos con el fin de determinar cuál acción ejecutar y cuando. A continuación, se propone establecer la lógica del sistema a partir de la realización de un modelo del sistema con una RdP, el monitor a través de la RdP determinará las acciones posibles de ejecución, como se muestra en la Fig. 1.

El sistema requiere de una política para la resolución de los conflictos de la RdP, este componente es utilizado para decidir entre las posibles acciones ejecutables. Además, es necesario un módulo para el manejo de los eventos y colas para almacenarlos (ver Fig. 1). Al descomponer el sistema reactivo (RS) o guiado por eventos (EDA) en estos componentes, se busca mejorar la gestión y el rendimiento delsistema embebido logrando simplificar su diseño, desacoplar los componentes y gestionar su control y ejecución en forma centralizada.

Los componentes agrupados en el área de puntos rojo (Fig. 1) corresponden a un requerimiento de cómputo y memoria importante (base de dato, lógica del sistema, comunicaciones, etc.) por lo que se ha adoptado una placa Raspberry PI para su implementación. En cambio, los agrupados en líneas de punto azul (Fig. 1) corresponden a componentes que realizan acciones como: abrir puertas, tomar imágenes, etc. Estos han sido implementados en una Espressif ESP-32, dada sucapacidad para manejar puertos de entrada salida, comunicaciones y bajo consumo. La arquitectura de bajo nivel resultante de la aplicación puede observarse en la Fig. 1, también se encuentra información detallada al respecto de su implementación en [7].

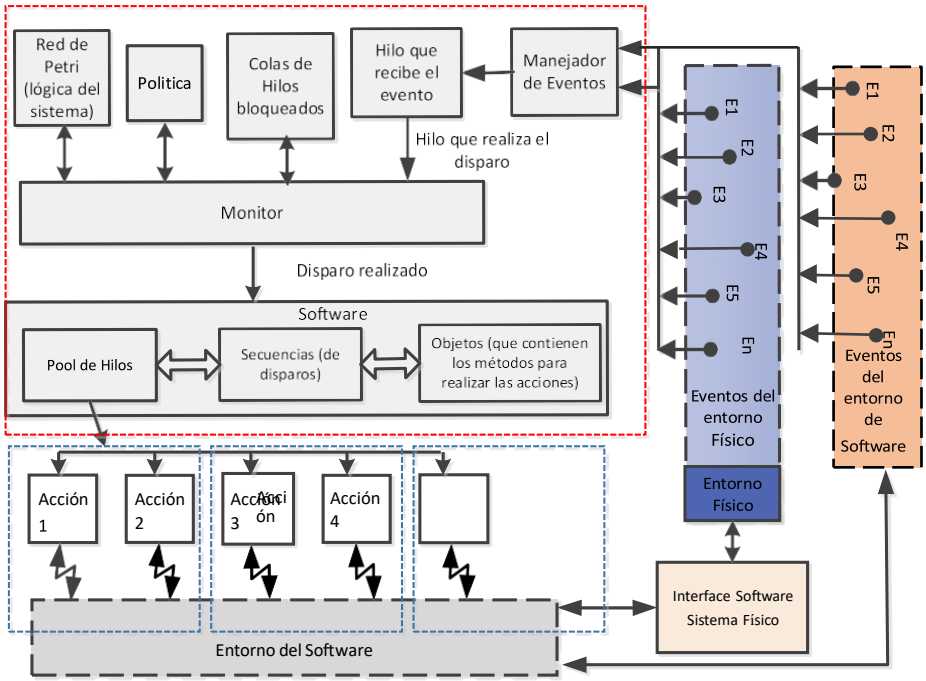


Fig. 1 Componentes de la arquitectura de bajo nivel

2.2 Diseño de componentes

En el diseño de la arquitectura se optó por la orientada a servicios por tratarse de una plataforma de recursos limitados como Raspberry Pi, principalmente en uso de memoria.

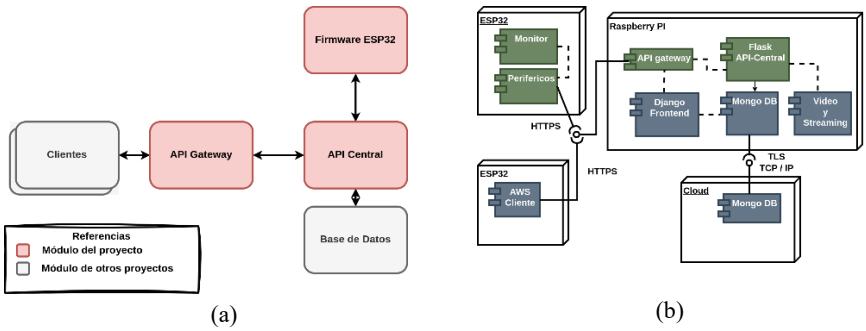
Para las comunicaciones, en una primera instancia, se usó el protocolo MQTT como mecanismo de comunicación entre los dispositivos ESP32-CAM y la Raspberry PI, por sus ventajas como sencillez y ligereza. Luego se descartó en favor de HTTP, ya que MQTT en Python sólo soporta comunicación asíncrona, lo que agrega complejidad y sobrecarga al diseño.

En general, Espressif [8] proporciona una placa de kit de desarrollo ESP32 que podemos usar directamente, este framework cuenta con una implementación de la librería pthreads. Esto se usó oportunamente para el desarrollo del monitor mediante diseños ya probados, los detalles de esta implementación se encuentran en [9]. La política para despertar los hilos de cada cola se delegó en el planificador de FreeRTOS.

También el framework provee implementaciones de clientes y servidores HTTP que fueron utilizados para construir la funcionalidad de cada endpoint.

En la Fig. 2 (a) se describe la arquitectura a nivel de módulos. El módulo API-Gateway sirve de punto de entrada para las consultas de los clientes, funciona como proxy reverso y centraliza la configuración de autenticación. Como parte de los mecanismos de seguridad, pese a que se implementan protocolos seguros en la REST API de los ESP32-CAM, los clientes nunca tienen acceso a ella, sino a través de la API central.

El diseño cuenta con una base de datos no relacional MongoDB [10] que se eligió debido al bajo acoplamiento de las entidades, la transparencia entre los diccionarios de Python y los documentos, sumado a la buena performance para almacenar archivos pequeños (imágenes).



(a) (b)

Fig. 2 Componentes de la arquitectura de alto nivel y diagrama de despliegue

Como parte del desacoplamiento y modularización de componentes, aquellos que se ejecutan en la plataforma Linux de Raspberry Pi fueron diseñados para ejecutarse en contenedores de Docker [11]. Con archivos de tipo 'Dockerfile' y 'docker-compose' se consigue que el sistema sea replicable, escalable y tolerante a fallos.

2.3 Topología de red

La Fig. 3 ilustra la topología de la red para el proyecto. Se divide en dos dominios: uno para el acceso de los usuarios a los clientes (web o REST) y otro para los dispositivos ESP32-CAM. La API central participa en ambos dominios gestionando las consultas. De esa manera se logró aislar el acceso a las interfaces expuestas por ellas y, a su vez, aumentar el alcance de la red inalámbrica.

La seguridad en los puntos de acceso, tanto en la API del nodo central como en cada dispositivo ESP32 consistió en la implementación del protocolo de Basic-Auth combinado con HTTPS [12] activando la verificación de certificados PKI.

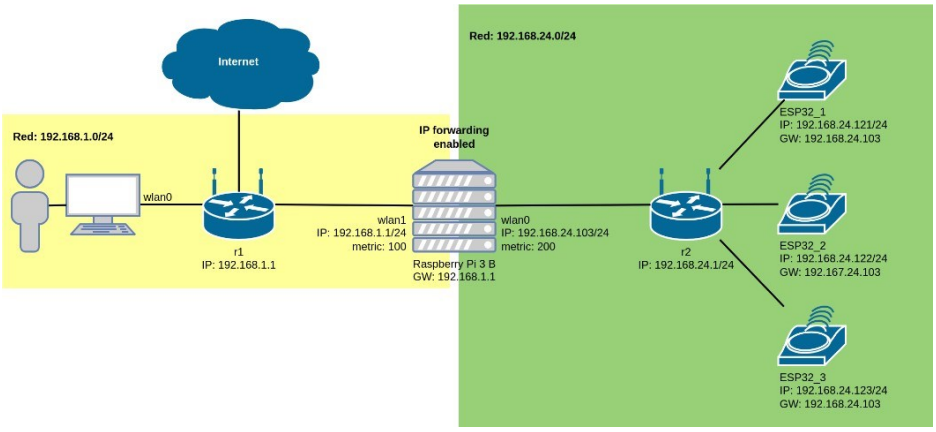


Fig. 3. Diseño topológico de las redes del sistema

2.4 Implementación y Selección de Componentes

A través de la utilización del *método iterativo,* se realizó el análisis, diseño, codificación y prueba de los módulos de software y la elección de componentes de hardware. Esto permitió obtener módulos funcionales al final de cada etapa, sirviendo los mismos como base para las siguientes iteraciones. La última de ellas ocurre cuando los requerimientos, repartidos estratégicamente, se cumplen.

Durante el desarrollo del sistema, luego de definir la arquitectura de bajo nivel, se procedió a realizar la selección de componentes de hardware. Para la implementación del hardware central se tuvieron en cuenta las siguientes opciones: Raspberry Pi, Orange Pi, Nvidia Jetson, Odroid, BeagleBone y Arduino. Luego de analizar las alternativas y con el criterio de menor costo, simplicidad de uso, soporte, y compatibilidad con los restantes módulos se seleccionó Raspberry Pi.

Para implementar el sistema de punto de acceso (puerta), se consideró la disponibilidad en el Laboratorio de Arq. de Computadoras de la F.C.E.F.y N. U.N.C. de la placa ESP32-CAM, que presenta diversas ventajas como su bajo costo, gran cantidad de puertos de entrada/salida, la disponibilidad de puertos genéricos y de vídeo. En cuanto a la elección del lenguaje de programación, se optó por Python debido a que es un lenguaje de alto nivel que posibilita un desarrollo rápido y eficiente. Aunquesi bien es cierto que su desempeño es más bajo comparado con otros lenguajes como

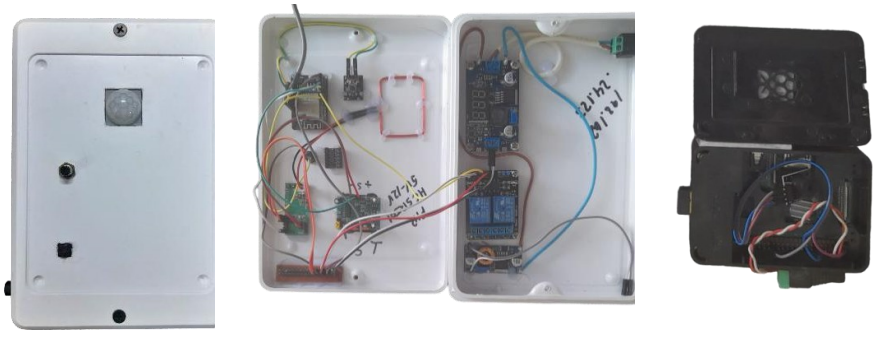
Java, C++ o Go, en este caso la velocidad no es la prioridad principal.

En lo referente a la selección del framework para el desarrollo de la API REST, se evaluaron las dos opciones más usadas para el desarrollo de aplicaciones web en Python, que son: Flask y Django. Para esta implementación se decidió utilizar Flask debido a que ofrece mayor flexibilidad y una curva de aprendizaje sencilla en comparación con Django. Para la selección del IDE/Framework luego de analizar las propiedades de Arduino, Espressif, Mongoose SO, Simba, Pumbaa y nanoFramework se optó por Espressif dado que implementa de forma nativa una versión de FreeRTOS compatible con multicore.

Para el periférico de acceso, se seleccionó RFID 125 KHz debido a su alta disponibilidad, lo que simplifica tanto su adquisición como reemplazo en caso de ser necesario.

En cuanto a fuente de alimentación DC-DC se optó por un modelo que implementa un voltímetro en la salida y la entrada con rango de tensión 1.23 – 35 V y 3A con un integrado LM2596. Los restantes periféricos fueron seleccionados de acuerdo con su disponibilidad y con criterio de menor costo con el objetivo de obtener un sistema final accesible y replicable.

La arquitectura del sistema de control de acceso distribuido se observa en la Fig. 4 (b), y fue orientada a servicios con protocolo http. La implementación del sistema para los puntos de acceso (puerta) se realizó en una carcasa genérica en donde se montaron los diferentes componentes.



(a) (b) (c)

Fig. 4. Vistas externa e interna del dispositivo de acceso e interna del dispositivo central.

El dispositivo resultante, en su vista exterior puede observarse en la Fig. 4 (a): en la parte superior el sensor de movimiento (HC-SR501), en el centro desplazado a

izquierda el pulsador y en la parte inferior el orificio por donde el módulo ESP32-CAM captura las imágenes.

En la Fig. 4 (b) se observa el interior de este dispositivo, conformado por: ficha de alimentación, fuente step-down DC-DC LM-2596 con voltímetro, conversor de niveles 5V - 3V3, placa ESP32-CAM con cámara OV2640 2Mp, módulo lector RFID RDM6300 y antena, módulo sensor de movimiento, interruptor, fichas y cables dupont. En la Fig. 4 (c) se puede observar la carcasa para el sistema central implementado con Placa SBC Raspberry PI 3b 1GB RAM 16GB SD y módulo buzzer pasivo.

El repositorio del proyecto se encuentra en https://github.com/orgs/Proyecto-Integrador-FCEFYN/dashboard.

3. Resultados

El presente trabajo expone como resultado un caso de estudio y aplicación, así como también la validación, de una metodología para diseñar y desarrollar sistemas embebidos, críticos, RS y EDA. Esta metodología logró gestionar eficazmente la complejidad del diseño, el testing y la codificación. Se minimizaron y mitigaron los riesgos desde el inicio.

Se obtuvo un sistema de control de acceso distribuido, capaz de permitir o restringir el acceso a zonas determinadas según parámetros de seguridad establecidos. Los usuarios son identificados mediante tarjetas o llaveros y se cuenta con datos personales almacenados al momento del registro. Además, el sistema cuenta con funcionalidades de seguridad adicionales tales como: restricciones horarias configurables, sensores de movimiento para la captura de imágenes ante situaciones inesperadas, grabación, almacenamiento y *streaming* de video, entre otras.

Entre las características del sistema se puede mencionar que es *distribuido* dado que esta implementado y soportado por múltiples dispositivos e interconectado mediante una red, *flexible* por ser apto para adaptarse a cambios en sus requerimientos minimizando su impacto por ser completamente modular, su lógica ha sido validada *formalmente* debido al fundamento matemático de las RdP. Es *mantenible* ya que es modificable efectiva y eficientemente debido a su modularidad, según necesidades evolutivas, correctivas o perfectivas y es *simple* por permitir visualizar las acciones, independientemente de la lógica y la política que conducen al sistema. Por lo tanto, se logra un código claro y sin responsabilidades solapadas.

Referencias

1. Tanenbaum, A.S. and M. Van Steen, Distributed systems: principles and paradigms. 2023: Prentice-Hall.
2. Hobbs, C., Embedded Software Development for Safety-Critical Systems. 2015: CRC Press.
3. Weilkiens, T., et al., Model-Based System Architecture. 2015: John Wiley & Sons.

4. Roberts, C.J., et al., Preliminary Results from a Model-Driven Architecture Methodology for Development of an Event-Driven Space Communications Service Concept. 2017.
5. Murata, T., Petri Nets: Properties, Analysis and Applications. Proceedings of the IEEE, 1989: p. Vol. 77, No. 4, pp. 541-580.
6. Ventre, L.O. and O. Micolini. Extended Petri Net Processor and Threads Quantity Determination Algorithm for Embedded Systems. in Argentine Congress of Computer Science. 2020. Springer.
7. Ventre, L.O. and O. Micolini. Algoritmos para determinar cantidad y responsabilidad de hilos en sistemas embebidos modelados con Redes de Petri S3PR. in XXVII Congreso Argentino de Ciencias de la Computación (CACIC)(Modalidad virtual, 4 AL 8 DE OCTUBRE DE 2021.). 2022.
8. Sharp, A. and Y. Vagapov, Comparative analysis and practical implementation of the ESP32 microcontroller module for the Internet of Things. 2022.
9. Melgarejo, M.ı.d.M.G. Comunicacion y Sincronizacion con Monitores Resumen del Tema. 2002 [cited 2023 Mayo]; Available from: http://www.lcc.uma.es/~gallardo/temaCLASE.pdf.
10. Sharma, M., Full Stack Development with MongoDB: Covers Backend, Frontend, APIs, and Mobile App Development Using PHP, NodeJS, ExpressJS, Python and React Native. 2022: BPB Publications.
11. Schenker, G.N., Learn Docker–Fundamentals of Docker 19. x: Build, test, ship, and run containers with Docker and Kubernetes. 2020: Packt Publishing Ltd.
12. mozilla.org. Autenticación HTTP. 2023 [cited 2023 Mayo]; Available from: https://developer.mozilla.org/es/docs/Web/HTTP/Authentication.

Coordinadores:

- Carlos Buckle (UNPSJB)
- Marcelo Arroyo (UNRC)
- Jorge Ardenghi (UNS)

CACiC 2023

CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

XVIII Workshop de Arquitecturas, Redes y Sistemas Operativos

> Short Papers

WARSO

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

CRANE: Simplificando el Despliegue de Aplicaciones Contenerizadas en Entornos Locales

Jose Miguel Silva Pavón[1], Franco Bellino[1], Patricia Bazán[3], Alejandra B. Lliteras[2,4], Nicolás del Rio[1]

[1] UNLP, Facultad de Informática, [2] UNLP Facultad de Informática, LIFIA, [3] UNLP, Facultad de Informática, LINTI, [4] CICPBA

js.silva.010@gmail.com, fran85bellino@gmail.com, pbaz@info.unlp.edu.ar, alejandra.lliteras@lifia.info.unlp.edu.ar, ndelrio@info.unlp.edu.ar

Abstract. CRANE es una herramienta diseñada para el despliegue local de aplicaciones en contenedores que busca simplificar las pruebas de entornos distribuidos de forma local. A diferencia de herramientas tradicionales como Minikube. El diseño de CRANE ofrece una solución liviana y de propósito general con capacidades de escalado automático, orientada a estudiantes y desarrolladores que requieran crear y desplegar stacks completos de aplicaciones. CRANE tiene como propósito, también, facilitar la incorporación de habilidades de *devops*, que acelera el proceso de desarrollo de software en un marco de entrega contínua.

En este trabajo, se presenta una API REST, construida para permitir la creación y el despliegue de servicios Docker, así como la monitorización, definición de políticas de escalado y gestión de alertas para la toma de decisiones.

Palabras Claves: Docker, Escalado, Monitorización, Api Rest, Devop, Minikube.

1. Introducción

En la era moderna de la tecnología, los servicios de Plataforma como Servicio (PaaS, por sus siglas en inglés Platform as a Service) han revolucionado el paradigma de desarrollo de software, proporcionando entornos completos de desarrollo y despliegue en la nube. Esta flexibilidad y comodidad han permitido a los desarrolladores centrarse en la lógica de aplicación, liberándose de las complejidades inherentes al manejo de la infraestructura. Sin embargo, cuando las necesidades no implican el uso de un PaaS, y es necesario trabajar con la infraestructura, surgen grandes inconvenientes:

- Migración del entorno a diferentes plataformas, cada una de ellas con diferentes características. Esto implica que haya archivos de configuración diferentes para cada entorno e instalación de dependencias propias donde se vaya a utilizar.
- Dificultad para llevar a cabo pruebas de rendimiento efectivas en etapas tempranas y construir aplicaciones verdaderamente escalables.
- La migración de aplicaciones entre hosts diferentes ha demostrado ser una tarea desafiante debido a las dependencias específicas del sistema, como las versiones de librerías y los sistemas operativos.

A raíz de estos desafíos, los contenedores de Docker han surgido como una solución prometedora, permitiendo el despliegue de aplicaciones en cualquier host con Docker instalado, independientemente de sus características específicas. Sin embargo, esto ha llevado a la necesidad de orquestar, medir y escalar estos contenedores de manera eficiente, un problema que las soluciones actuales como Kubernetes, han podido resolver pero con un alto costo en recursos computacionales. Asimismo, la configuración de estos contenedores frecuentemente requiere la creación de archivos de configuración estáticos que dependen de rutas locales, lo que complica aún más el proceso.

Debido a estas necesidades, surgen nuevos roles, como el de DevOps (Development and Operations) que une roles que anteriormente estaban separados (desarrollo, operaciones de TI, ingeniería de la calidad y seguridad) en un solo rol y así tener una visión completa de todo el sistema a desarrollar.

En este contexto, se propone una innovación en la herramienta CRANE, originalmente presentada por [Arcidiacono et al., 2022], para mejorar el proceso de despliegue y control de servicios.

CRANE es un diseño de solución que se compone de un Front-End y un Back-End. El presente trabajo se basa en la parte Back-End de CRANE.

Esta propuesta abarca los siguientes puntos:

- Una API para la gestión de los contenedores, por lo tanto la interacción entre el usuario y la herramienta es transparente e independiente de donde se ejecuta.
- La medición del estado de los contenedores está integrada en la herramienta, y en caso de que esa medición arroje la necesidad de instanciar contenedores, esta se realiza automáticamente.

Este documento se organiza de la siguiente manera: en la sección 2 se enumeran las metodologías de despliegue de aplicaciones actuales, así como sus características principales. En la sección 3 se analiza el diseño de CRANE y qué lugar ocupa en el despliegue actual de aplicaciones. En la sección 4 se detalla la implementación técnica y que herramientas se utilizaron para su desarrollo. En la sección 5 se explica que futuras implementaciones se pueden realizar a partir de CRANE.

2. Conceptos vinculados al despliegue de aplicaciones

El despliegue de aplicaciones es una fase crítica en el ciclo de desarrollo de software, ya que pone en funcionamiento en entornos de producción, lo desarrollado. En los últimos años ha habido un avance en las técnicas y en las herramientas usadas para que este proceso se pueda realizar de una manera rápida, eficiente y confiable.

A continuación se listan algunas de ellas:

Despliegue Continuo: El despliegue continuo consiste en la automatización del proceso de entrega e implementación de código en producción de forma constante y rápida. Esta metodología se adopta para liberar actualizaciones en producción con mayor frecuencia, lo que permite una “mejora continua” de la aplicación y mayor respuesta.

Contenedores: La tecnología de contenedores popularizada por Docker, ha revolucionado el despliegue de aplicaciones, ya que permite empaquetar todo el entorno de ejecución de la aplicación y sus dependencias en un contenedor aislado. Esto nos garantiza portabilidad y consistencia de las aplicaciones en diferentes entornos y es el tipo en el que se basa CRANE.

Microservicios: Los microservicios son una arquitectura de diseño de software donde las aplicaciones se componen de pequeños servicios independientes que se comunican entre sí, mediante interfaces bien definidas. Esto nos permite poder desplegar cada servicio de manera independiente, lo que facilita la escalabilidad y mantenimiento de la aplicación.

Infraestructura como código (IaC): IaC es una práctica en la que toda la infraestructura necesaria para ejecutar una aplicación se define y se gestiona mediante código. Herramientas como Terraform y Ansible permiten crear y configurar recursos de infraestructura en la nube de forma automatizada.

Monitoreo: El monitoreo de aplicaciones es fundamental para asegurar un despliegue exitoso y la posterior corrección de errores, usando mecanismos de monitoreo en tiempo real.

Servicios en la nube: El uso de servicios en la nube, ha simplificado el despliegue de aplicaciones. Servicios como AWS, AZURE y Google Cloud ofrecen plataformas con herramientas que facilitan el despliegue y gestión de aplicaciones, reduciendo la complejidad.

3. El diseño de CRANE para el proceso de despliegue de aplicaciones

El diseño de CRANE propuesto en este trabajo se enfoca en mejorar el proceso de despliegue de aplicaciones contenerizadas, así como su monitoreo y escalado, haciendo énfasis en la transparencia y facilidad de uso para el usuario. CRANE se basa en la implementación de dos componentes principales: un Back-End para la creación y despliegue de contenedores Docker, y un Front-End destinado al usuario final.

El componente de Back-End, utiliza Python y FastAPI[1] para proporcionar una API que permite la gestión de contenedores Docker. Esto incluye la creación, despliegue y monitoreo de contenedores, así como la definición de políticas de escalado y la gestión de alertas.

El componente de Front-End, a través de la API de CRANE, permite al usuario interactuar con la herramienta de forma transparente e independiente del lugar donde se ejecute. La interacción con el usuario se realiza mediante una interfaz web intuitiva

[1] https://fastapi.tiangolo.com/

y fácil de usar, diseñada para simplificar la experiencia del usuario y reducir la curva de aprendizaje.

Para el monitoreo y escalado de los contenedores, CRANE utiliza Prometheus[2] para recoger métricas de rendimiento y Alert Manager para gestionar alertas basadas en esas métricas. Además, para el balanceo de carga entre los contenedores y el enrutamiento de las solicitudes, se utiliza Traefik[3]. De esta forma, CRANE proporciona un entorno de despliegue completo, incluyendo balanceador, métricas, alertas y más, como se puede ver en la arquitectura mostrada en la Figura 1.

Todo esto se encapsula en un Docker Compose que se genera automáticamente para cada despliegue, lo que permite a CRANE desplegar un stack totalmente configurado de manera automática y sin necesidad de intervención manual por parte del usuario.

4. CRANE como servicio: tecnologías seleccionadas

La primera versión de CRANE se basaba en el uso de un script de consola en bash que empleaba comandos Docker para instanciar contenedores, así como para escalar y desescalar. Sin embargo, esto exigía que el administrador tomara decisiones continuamente sobre cuándo escalar o desescalar una instancia. Para superar esta limitación, se desarrolló la segunda versión de CRANE, capaz de escalar automáticamente mediante métricas de uso, tales como la cantidad de peticiones, tiempos de respuesta, uso de CPU o memoria, en una forma similar a Keda[4] en Kubernetes.

A pesar de esta mejora, la segunda versión tenía el inconveniente de estar vinculada al sistema operativo en el que se usara. Por lo tanto, se desarrolló una tercera versión, con una API REST y una interfaz gráfica para interactuar con Docker mediante mensajes HTTP.

Para este trabajo, se utilizó como punto de partida esta tercera versión de CRANE. La arquitectura se organizó en torno a un Proxy, como se puede ver en la Figura 1, que recibe peticiones y balancea la carga en función de métricas recogidas por Prometheus y Alert Manager, además de actuar de acuerdo a políticas preconfiguradas mediante Open Policy. Todo esto se configuró a través de una API REST creada en Python.

Se optó por utilizar Traefik [Sharma, R. y Mathur, A.,2021] en lugar de Nginx como proxy debido a su configuración y gestión simplificada, así como su capacidad para detectar automáticamente nuevos servicios o contenedores y ajustar su configuración, lo cual simplifica considerablemente la administración de la infraestructura.

[2] https://prometheus.io/

[3] https://doc.traefik.io/traefik/

[4] https://keda.sh/

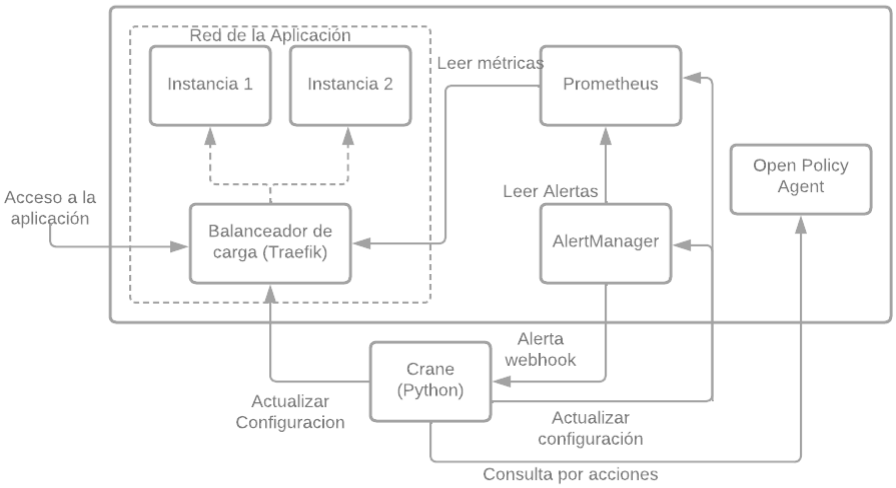


Fig. 1. Tercera versión de la arquitectura CRANE

En el proceso de este trabajo, se pusieron de manifiesto diversos desafíos. El aprendizaje de fastAPI, la comprensión de cómo conectar todas las piezas y la interacción con Docker desde Python ya que presentaron dificultades iniciales. La complejidad de CRANE al generar entornos completos para desplegar servicios exigió un esfuerzo adicional para asegurar que todo encajara de forma automática.

Para solucionar estos inconvenientes se analizó la documentación de fastAPI y Docker, así como de todo su entorno relacionado, recurriendo a la ayuda de la comunidad y experimentando con diferentes configuraciones hasta lograr una integración adecuada. La instanciación de contenedores Whoami y la validación a través de su URL permitieron confirmar que cada vez se devolvía una dirección IP diferente, asegurando que los contenedores fueron instanciados y que la carga se repartía entre ellos.

Las lecciones aprendidas incluyen la importancia de una planificación cuidadosa, la necesidad de un profundo conocimiento en las herramientas utilizadas y la comprensión de que la experimentación y la perseverancia pueden superar los obstáculos técnicos.

Este proyecto permitió una comprensión profunda de la orquestación de contenedores y brindó la posibilidad de desarrollar habilidades valiosas en la automatización y escalado de infraestructuras.

5. Conclusiones y Trabajo Futuro

En este artículo, se presentó una forma de automatizar el despliegue de aplicaciones contenerizadas y posterior gestión de métricas, de forma que sea transparente al usuario. Primero se partió del esquema definido en CRANE y se comenzó a conectar las diferentes partes que componen una simple aplicación. Una vez instanciada la app se

implementó una API Rest que levanta la app, solo haciendo una llamada a un endpoint. Se obtuvo un resumen de lecciones aprendidas, que pueden ser reusadas en personas que se sumen a proyectos de estas características. Como trabajo futuro se pretende implementar una interfaz web usando React que pueda funcionar con CRANE de una manera totalmente visual y disminuir la curva de aprendizaje de la app por parte de los usuarios.

Referencias

1. Arcidiacono, J., Bazán, P., del Río, N., & Lliteras, A. B. (2022). CRANE: A Local Deployment Tool for Containerized Applications. In Conference on Cloud Computing, Big Data & Emerging Topics (pp. 58-71). Springer, Cham.
2. Arcidiacono, J., Lliteras, A. B., & Bazán, P. A. (2020). DEHIA, una plataforma para la generación y ejecución de actividades de recolección de datos con intervención humana aplicada en el Programa E-Basura. In VI Simposio Argentino de Ciencia de Datos y GRANdes DAtos (AGRANDA 2020)-JAIIO 49 (Modalidad virtual).
3. Bullington-McGuire, R. and Dennis, A.K. and Schwartz, M. (2020). Docker for Developers: Develop and run your application with Docker containers using DevOps tools for continuous delivery. Packt Publishing.
4. Burns, B., Beda, J., Hightower, K., & Evenson, L. (2022). Kubernetes: up and running. " O'Reilly Media, Inc.".
5. Ferraiolo, D., Cugini, J., & Kuhn, D. R. (1995, December). Role-based access control (RBAC): Features and motivations. In Proceedings of 11th annual computer security application conference (pp. 241-48).
6. Fielding, R. T., & Taylor, R. N. (2002). Principled design of the modern web architecture. ACM Transactions on Internet Technology (TOIT), 2(2), 115-150.
7. Httermann, M. (2012). DevOps for developers. Apress: delivers a practical, thorough introduction to approaches, processes and tools to foster collaboration between software development and operations
8. Rad, B. B., Bhatti, H. J., & Ahmadi, M. (2017). An introduction to docker and analysis of its performance. International Journal of Computer Science and Network Security (IJCSNS), 17(3), 228.
9. Rani, D., & Ranjan, R. K. (2014). A comparative study of SaaS, PaaS and IaaS in cloud computing. International Journal of Advanced Research in Computer Science and Software Engineering, 4(6).
10. Reis, D., Piedade, B., Correia, F. F., Dias, J. P., & Aguiar, A. (2021). Developing docker and docker-compose specifications: A developers' survey. IEEE Access, 10, 2318-2329.
11. Sommerlad, P. (2003, June). Reverse Proxy Patterns. In EuroPLoP (pp. 431-458).
12. Sharma, R., Mathur, A., Sharma, R., & Mathur, A. (2021). Traefik for Microservices. Traefik API Gateway for Microservices: With Java and Python Microservices Deployed in Kubernetes, 159-190.

Coordinadores:

- Guillermo Leguizamón (UNSL)
- Carolina Salto (UNLPam)
- Waldo Hasperué (UNLP)

CACiC 2023

CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

XXIV Workshop de Agentes y Sistemas Inteligentes

> Full Papers

WASI

Generación de gestos de lengua de señas con redes neuronales generativas basadas en poses y etiquetas

Gaston Gustavo Rios[1,3] [0000-0003-0252-7036], Pedro Dal Bianco[1,3] [0000-0001-7197-8602], Franco Ronchetti[1,2][0000-0003-3173-1327], Facundo Quiroga[1] [0000-0003-4495-4327], Oscar Stanchi[1,3] [0000-0003-0294-2053], and Waldo Hasperué[1,2] [0000-0002-9950-1563]

[1] Instituto de Investigación en Informática LIDI - Universidad Nacional de La Plata., La Plata, Argentina
{grios,pdalbianco,fronchetti,fquiroga, ostanchi,whasperue}@lidi.info.unlp.edu.ar
https://weblidi.info.unlp.edu.ar

[2] Comisión de Investigaciones Científicas de la Pcia. de Bs. As. (CIC-PBA). Argentina
https://www.cic.gba.gob.ar

[3] Becario de postgrado UNLP
https://unlp.edu.ar

Abstract. Obtener datos etiquetados para el entrenamiento de redes neuronales en tareas de reconocimiento de lengua de señas es un desafío difícil y costoso. En este artículo investigamos la factibilidad de generar datos utilizando Generative Adversarial Networks (GAN), para mejorar el entrenamiento de redes neuronales. Específicamente, generamos imágenes de manos condicionando los modelos GAN con información semántica de poses y etiquetas. Comparamos los modelos ReACGAN y SPADE en la generación de nuevas imágenes de alta calidad. Evaluamos la generación de señas en dos conjuntos de datos: RWTH y HaGRID. Se entrenaron modelos generativos utilizando subconjuntos de tamaño reducido para probar el efecto de la reducción de datos de entrenamiento. Medimos la calidad de los modelos resultantes utilizando métricas cuantitativas (FID, IS, cobertura y densidad) y cualitativas (encuestas). Como resultado obtuvimos modelos GAN capaces de generar señas con un buen nivel de realismo que luego podrán ser utilizados para aumentar conjuntos de datos de lengua de señas.

Keywords: Reconocimiento de señas · Lengua de señas · Red generativa antagónica · SPADE · ACGAN · Estimación de pose.

1. Introducción

En los últimos años, el rendimiento de los modelos de aprendizaje profundo ha aumentado considerablemente. Sin embargo, el rendimiento de los modelos está estrechamente relacionado con la calidad del conjunto de datos utilizado para entrenarlos, lo que requiere conjuntos de datos grandes con suficientes muestras para aproximar la verdadera distribución del dominio. Además, la creación de nuevos conjuntos de datos etiquetados es una tarea difícil y costosa, que a menudo requiere la participación de expertos durante la recopilación y etiquetado de los datos [3]. Este es el caso de los conjuntos de datos de lenguas de señas, donde se necesita un intérprete para etiquetar los datos existentes o grabar nuevas muestras con precisión[4].

Dado que las lenguas de señas no son mutuamente inteligibles, cada una requiere su propio conjunto de datos [3]. Debido a la dificultad de la tarea en la creación de

conjuntos de datos de lenguas de señas, las comunidades con menos recursos se ven limitadas en el tamaño y la calidad de sus conjuntos de datos. Incluso las comunidades con recursos abundantes tienen conjuntos de datos de lengua de señas limitados que no son adecuados para entrenar modelos. La falta de datos dificulta el entrenamiento de modelos que se puedan aplicar a un escenario real en esta tarea. Para nuestros experimentos utilizamos los conjuntos de datos RWTH y HaGRID. RWTH es un conjunto de lengua de señas alemanas tomado de un canal de noticias del clima. Presenta poca variación de señantes y señas, imágenes borrosas y una cantidad limitada de datos etiquetados. En contraste, HaGRID es un conjunto de datos de gestos de gran tamaño creado para entrenar modelos detectores de gestos. HaGRID por lo tanto es un buen punto de comparación de la capacidad de los modelos de generar imágenes de alta calidad.

En este artículo evaluamos diferentes métodos para entrenar modelos generativos condicionados por poses y etiquetas. Entrenamos las arquitecturas de modelos GAN: ReACGAN y SPADE. Adicionalmente evaluamos diferentes pesos para la parte condicional de la función de pérdida, lo que nos llevó a la decisión de entrenar nuestros modelos con un alto peso condicional, aumentando el condicionamiento de las imágenes para que no se alejen de la distribución de las clases sin perder calidad. Comparamos el uso de las poses de las manos extraídas con OpenPose como líneas 2D en las uniones de las articulaciones o puntos en cada articulación. Realizamos experimentos utilizando los 2 métodos en RWTH llegando a resultados similares, por lo que decidimos utilizar las uniones para entrenar los modelos en HaGRID. Como resultado obtuvimos modelos generadores capaces de generar, a partir de etiquetas o poses, nuevas imágenes de gestos de manos con un alto nivel de realismo.

2. Marco Teórico

2.1 Modelos generativos

Recientemente, los modelos generativos han mostrado grandes mejoras en la calidad de las imágenes sintéticas. Los modelos más exitosos, como las Redes Generativas Antagónicas (GAN) [7], los Autoencoders Variacionales (VAE) [12] y los Modelos de Difusión[8], pueden generar nuevas imágenes realistas sin memorizar los datos del conjunto de entrenamiento. Con estos modelos, podemos generar un número arbitrario de nuevas muestras de datos. Sin embargo, estas imágenes pueden presentar artefactos que agregan ruido al entrenar nuevos modelos con ellas. Además, el colapso de moda puede afectar la variación de las imágenes generadas, lo que resulta en una cantidad limitada de imágenes únicas[1].

Redes Generativas Antagónicas (GAN) Las GAN funciona entrenando de manera conjunta un discriminador y un generador, donde el generador minimiza la distancia entre las distribuciones de los datos generados y reales con el objetivo de que el discriminador no pueda distinguirlos. Por otro lado, el discriminador es entrenado para indicar si los datos son reales o falsos [6]. De esta forma, ambos modelos son entrenados hasta alcanzar un equilibrio de Nash. Esto se dá cuando, dados los parámetros actuales de ambos modelos, la función de costo de cada modelo llega a un mínimo local.

2.2 Generación condicionada

Las GAN no condicionales no tienen forma de controlar la moda de los datos generados, ya que estos son generados a partir de un vector de ruido *z*. Las Conditional Generative Adversarial Nets (CGAN) [15] introducen la idea de utilizar datos adicionales *y* para de esta forma condicionar los resultados del generador. Estos datos adicionales serán utilizados tanto en el generador como en el discriminador para condicionar el entrenamiento. Este tipo de modelo permite la utilización de etiquetas o de otros tipos de datos (por ejemplo, texto utilizando *embeddings* para codificar la entrada de texto generando vectores similares para conceptos relacionados) como condicionador. ReACGAN [9] se propuso como una mejora de los métodos utilizados por Auxiliary Classifier GAN (ACGAN). ACGAN utiliza información condicional durante el entrenamiento, incluyendo en este una pérdida de clasificación adicional a la pérdida del modelo, lo que aumentó su rendimiento en comparación con GAN regular. Sin embargo, ACGAN ha mostrado tener un entrenamiento inestable cuando el número de clases aumenta y colapsa a una cantidad pequeña de datos generados fácilmente clasificables. Estos problemas se abordaron en ReACGAN al proyectar los vectores de entrada sobre una hiperesfera unitaria y al utilizar comparaciones de dato a dato en cada minibatch. Al hacer esto, ReACGAN logró resultados estado del arte en modelos GAN y obtuvo un mejor rendimiento que algunos modelos de difusión. SPADE [17] es una GAN condicional que originalmente se diseñó para usar una capa de segmentación para condicionar la generación de nuevos datos sintéticos. Introduce un nuevo método de normalización similar al módulo de Batch Normalization que permite el uso de datos 2D mediante convoluciones.

2.3 Generación de lengua de señas

La generación de lengua de señas es una tarea que ha realizado avances recientemente con la maduración de los modelos de redes neuronales generativas. La generación de señas realistas provee una nueva forma de agilizar la interacción entre personas señantes y no señantes, y el aprendizaje de las diversas lenguas. También este tipo de modelos permitirán utilizar los nuevos datos creados para entrenar otro tipo de modelos con conjuntos de datos de lengua de seña con datos etiquetados limitados. La tarea de la generación de señas de la lengua de señas ha sido enfrentada anteriormente mediante la generación de poses [24,20], basada en avatares[11] y de imágenes foto-realistas [21,23,16]. En particular, la generación de imágenes foto-realistas ha mostrado que los modelos GAN son un método efectivo para lograr crear imágenes de personas comunicándose mediante lengua de señas con un buen nivel de realismo.

3. Métodos Propuestos

Elegimos GAN en lugar de los Modelos de Difusión debido a su velocidad de inferencia más rápida y similar rendimiento, lo que permite la generación de grandes conjuntos de datos de imágenes sintéticas sobre la marcha. Esta capacidad puede ser beneficiosa para reducir dinámicamente el sobreajuste, similar a los beneficios del aprendizaje activo [19]. Sin embargo, los modelos GAN convencionales son propensos

al colapso de moda, lo que disminuye la diversidad de las imágenes generadas. Además, el entrenamiento de estos modelos es inestable y requiere muchas soluciones alternativas para entrenarlos de manera efectiva. Un método conocido para estabilizar el entrenamiento y reducir el colapso de moda es agregar información condicional a los datos de entrenamiento. Esto incluye a los modelos ReACGAN condicionados por etiquetas y SPADE, que condiciona la salida a través de un mapa semántico.

En los experimentos realizados utilizamos Generative Adversarial Networks (GAN) condicionados por etiquetas, puntos clave de la mano y las uniones entre los puntos clave. Para condicionar a partir de la etiqueta 1 utilizamos un clasificador auxiliar [9], mientras que para condicionar en los puntos clave y las uniones entre ellos 2 se utilizaron las capas de normalización SPatially-Adaptive (DE) normalization (SPADE) [17].

4. Experimentos realizados

En esta sección detallamos los conjuntos de datos utilizados, la metodología utilizada para realizar los entrenamientos de los modelos generativos y los resultados obtenidos. Cada modelo fue evaluado utilizando las métricas Fréchet Inception Distance (FID), Inception Score (IS), Cobertura y Densidad.

4.1 Conjuntos de Datos

RWTH [13] está compuesto por 3359 imágenes etiquetadas y 1 millón débilmente etiquetadas de señales capturadas del canal de televisión alemán PHOENIX. Las señales pertenecen a la lengua de señas alemana. Todas las imágenes están recortadas centradas en los señantes. El conjunto de datos contiene un total de 45 señas diferentes. El conjunto de datos está altamente desequilibrado y presenta una gran variación intraclase y similitud entre diferentes clases. Los intérpretes siempre visten ropa negra sobre un fondo blanco.

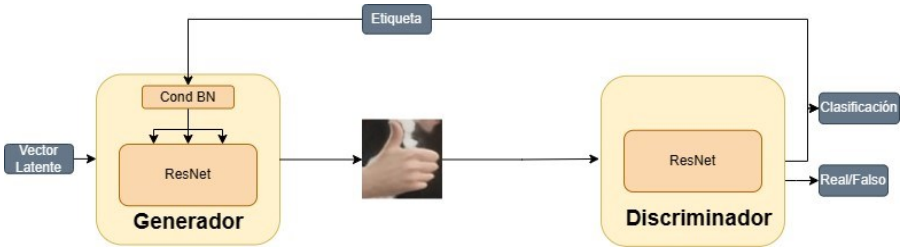


Fig.1: Diagrama del modelo ReACGAN. El generador toma como entrada un vector latente muestreado de una distribución gauseana y una etiqueta. El discriminador toma como entrada una imagen generada o real. El discriminador tiene dos salidas que son utilizadas para calcular la perdida *Data-to-Data CrossEntropy* (D2D-CE) y antagónica.

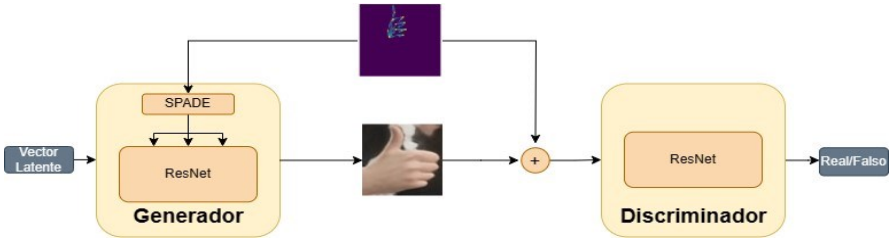
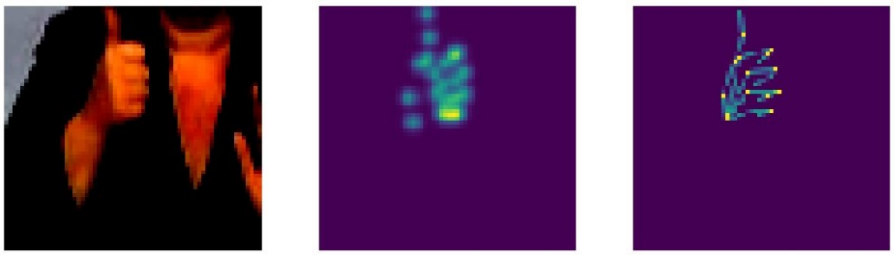


Fig.2: Diagrama del modelo SPADE. El generador toma como entrada un vector latente muestreado de una distribución gauseana y una pose codificada en *c* canales con un tamaño idéntico a la imagen de salida. El discriminador tomara como entrada la unión a nivel de los canales entre una imagen falsa o verdadera y su respectiva pose. Posteriormente utilizara la salida para calcular la perdida antagónica.

HaGRID - HAnd Gesture Recognition Image Dataset [10] fue creado para la clasificación y detección de gestos estáticos de mano. HaGRID consta de 552,992 imágenes RGB FullHD de cuerpo completo con 18 clases de gestos de mano y una clase "sin gesto". Hay un total de 34,730 personas únicas con al menos la misma cantidad de escenas. HaGRID muestra una alta diversidad entre cada persona, iluminación y fondo. El conjunto de datos también proporciona las poses para cada mano.



(a) Original (b) Puntos clave (c) Uniones

Fig.3: Visualización de la imagen original, sus puntos clave y uniones. Cada punto clave es representado como un canal con una distribución normal multivariada centrada en la coordenada del punto clave. Las uniones consisten en una línea por canal que une dos puntos anatómicamente adyacentes.

4.2 Preprocesamiento

Para obtener las poses de las señas de RWTH utilizamos OpenPose [5,22]. Esto resultó en un total de 21 puntos clave por mano, pero dado que trabajamos con señas de una sola mano, solo utilizamos la mano con mayor confianza en cada imagen. Eliminamos las muestras para las cuales no pudimos extraer ninguna pose, reduciendo el total de imágenes de 3359 a 2098.

En el caso de HaGRID, decidimos recortar las manos en las imágenes para obtener una mejor precisión en la generación especifica de estas, por lo que solo trabajamos con las manos de las imágenes ignorando el resto del cuerpo.

Para ambos conjuntos de datos modificamos el tamaño de cada imagen a 64x64 y normalizamos los valores de los píxeles. Para el aumento de datos incluimos volteo, translación, *cutout* y cambios en el brillo, saturación y contraste de las imágenes. Aislamos algunas muestras de cada conjunto de datos para usarlas como conjunto de prueba para nuestro clasificador. El resto de los datos se utilizó para el entrenamiento y la validación.

Las poses son representadas como un mapa de características $P \in \mathrm{R}^{cx64x64}$ donde c es la cantidad de canales. En el condicionamiento por pose utilizando puntos clave, cada canal posee una distribución normal multivariada centrada en la coordenada de un punto. Por lo tanto, este tipo de entrada posee un total de 21 canales. En el caso del condicionamiento utilizando las líneas de las uniones entre puntos se dibuja una línea por canal, creando un total de 20 canales.

4.3 Configuración de las evaluaciones

Empleamos la misma arquitectura base para todos los modelos GAN y aplicamos Normalización espectral para estabilizar el entrenamiento. Para condicionar los modelos con las etiquetas, utilizamos Conditional Batch Normalization en ReACGAN, y, alternativamente, utilizamos módulos SPADE para condicionar el modelo con los puntos. En todas las configuraciones utilizamos perdidas bisagra [14] y asignamos una pérdida condicional alta para mejorar la capacidad del modelo de generar imágenes con la etiqueta o pose correcta, ya que una pérdida condicional alta demostró aumentar la calidad de las imágenes generadas.



(a) RWTH real



(b) HaGRID real



(c) RWTH etiqueta



(d) HaGRID etiqueta



(e) RWTH pose



(f) HaGRID pose

Fig.4: Ejemplos reales y generados de RWTH y HaGRID con los modelos ReACGAN y SPADE condicionados por etiqueta y datos de pose respectivamente.

Table 1: Comparación de la performance de los modelos GAN en el conjunto de datos RWTH. La tabla muestra los resultados de la evaluación de los modelos utilizando las métricas Fréchet Inception Distance (FID), Inception Score (IS), Cobertura y Densidad.

RWTH GAN	**FID(↓)**	**IS(↑)**	**Cobertura(↑)**	**Densidad(↑)**
ReACGAN λ_{cond}=0.5	45.45	2.21	0.52	0.33
ReACGAN λ_{cond}=1	**45.19**	2.11	**0.60**	**0.48**
SPADE-puntos-clave	51.96	2.24	0.43	0.30
SPADE-uniones	51.05	**2.25**	0.38	0.23

Table 2: Comparación de la performance de los modelos GAN en el conjunto de datos HaGRID. La tabla muestra los resultados de la evaluación de los modelos utilizando las métricas cuantitativas Fréchet Inception Distance (FID), Inception Score (IS), Cobertura y Densidad. También se muestra como métrica cualitativa con participantes humanos la media del nivel de realismo basada en encuestas con una puntuación del 1 al 5.

HaGRID GAN	**FID(↓)**	**IS(↑)**	**Cobertura(↑)**	**Densidad(↑)**	**Humano**
ReACGAN	13.9	3.62	0.88	0.80	3.97
SPADE-uniones	33.21	3.88	0.65	0.70	-

4.4 Resultados

De cada modelo medimos su FID e IS, dos métricas estándar para medir la calidad de las imágenes generadas de los modelos generativos [2]. Ambas métricas están basadas en la utilización del modelo Inception, de ahí proviene sus nombres, y proveen un método cuantitativo para medir el realismo de las imágenes basado en la distancia entre la distribución de datos reales y generados. Por otro lado, estas métricas tienen sus limitaciones, por lo que se agregaron las métricas de cobertura y densidad, las cuales permiten medir por separado la diversidad y fidelidad de las imágenes respectivamente. La fidelidad indica el grado de similaridad entre las imágenes generadas y las reales. La diversidad, por otro lado, mide si los elementos generados

cubren la totalidad de la variabilidad de los elementos reales. También se realizaron pruebas cualitativas para el modelo ReACGAN entrenado con HaGRID por medio de cuestionarios donde los participantes fueron dados la tarea de puntuar las imágenes generadas del 1 al 5 en 3 áreas: realismo de la mano, correcta anatomía de los dedos y correcta posición de los dedos. Como resultado de estas pruebas cualitativas se obtuvo una media de 3.97 en cuanto al realismo de las imágenes, 3.70 respecto a la anatomía de los dedos y 4.03 respecto a la posición de los dedos.

Las Tablas 1 y 2 muestran el rendimiento de los diferentes modelos GAN entrenados en los conjuntos de datos RWTH y HaGRID. Aumentar el peso de la pérdida condicional genera una ligera mejora en las métricas del modelo generador. Sin embargo, no hubo una diferencia clara en el rendimiento de los modelos condicionados en puntos clave o en las líneas de las manos. Los modelos condicionados por etiqueta mostraron una mejor calidad, obteniendo mejores métricas que los modelos condicionados por pose en todos los casos.

5. Conclusiones

En este articulo realizamos la generación de manos utilizando modelos con arquitecturas y métodos de condicionamiento variados. Como resultado, se obtuvieron modelos capaces de generar imágenes con un buen nivel de realismo, lo cual fue validado mediante métricas cualitativas y cuantitativas. Los modelos condicionados por etiquetas obtuvieron un mejor resultado que los condicionados en pose, esto puede deberse a la utilización de D2D-CE para calcular la perdida en ReACGAN.

Los modelos generadores creados son capaces de generar nuevos conjuntos de datos sintéticos de lengua de seña que luego podrían ser utilizados para entrenar otro tipo de modelos. Esto permitiría el entrenamiento de modelos con menor cantidad o variabilidad de datos, un problema común en conjuntos de datos de la lengua de señas. A futuro, sería interesante la búsqueda de una mejor pérdida condicional para el modelo condicionado en poses. De esta forma, dada la semejanza de los dominios de las señas y los gestos, sería posible la realización de una generación *zero-shot*[18] de RWTH con un modelo entrenado condicionado con poses en HaGRID.

Referencias

1. Arora, S., Zhang, Y.: Do gans actually learn the distribution? an empirical study. CoRR abs/1706.08224 (2017)
2. Borji, A.: Pros and cons of gan evaluation measures (2018)
3. Bragg, D., Caselli, N., Hochgesang, J.A., et al.: The fate landscape of sign language ai datasets: An interdisciplinary perspective. ACM Trans. Access. Comput. 14(2) (2021). https://doi.org/10.1145/3436996
4. Bragg, D., Koller, O., Bellard, M., et al.: Sign language recognition, generation, and translation: An interdisciplinary perspective (2019)
5. Cao, Z., Hidalgo Martinez, G., Simon, T., et al.: Openpose: Realtime multi-person 2d pose estimation using part affinity fields. IEEE Transactions on Pattern Analysis and Machine Intelligence (2019)

6. Goodfellow, I.J.: NIPS 2016 tutorial: Generative adversarial networks. CoRR abs/1701.00160 (2017)
7. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., et al.: Generative Adversarial Networks. arXiv e-prints p. arXiv:1406.2661 (2014).
8. https://doi.org/10.48550/arXiv.1406.2661
9. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. CoRR abs/2006.11239 (2020)
10. Kang, M., Shim, W., Cho, M., et al.: Rebooting ACGAN: auxiliary classifier gans with stable training. CoRR abs/2111.01118 (2021)
11. Kapitanov, A., Makhlyarchuk, A., Kvanchiani, K.: HaGRID - HAnd Gesture Recognition Image Dataset. arXiv e-prints p. arXiv:2206.08219 (2022)
12. Kim, J.H., Hwang, E.J., Cho, S., et al.: Sign language production with avatar layering: A critical use case over rare words. In: Proceedings of the Thirteenth Language Resources and Evaluation Conference. pp. 1519–1528. European Language Resources Association, Marseille, France (2022)
13. Kingma, D.P., Welling, M.: Auto-encoding variational bayes (2022)
14. Koller, O., Ney, H., Bowden, R.: Deep hand: How to train a cnn on 1 million hand images when your data is continuous and weakly labelled. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3793–3802
15. (2016). https://doi.org/10.1109/cvpr.2016.412
16. Lim, J.H., Ye, J.C.: Geometric gan (2017)
17. Mirza, M., Osindero, S.: Conditional generative adversarial nets. CoRR abs/1411.1784 (2014)
18. Natarajan, B.S., R., E.: Dynamic gan for high-quality sign language video generation from skeletal poses using generative adversarial networks. Soft Computing 26, 13153–13175 (2021)
19. Park, T., Liu, M., Wang, T., et al.: Semantic image synthesis with spatiallyadaptive normalization. CoRR abs/1903.07291 (2019)
20. Ramesh, A., Pavlov, M., Goh, G., et al.: Zero-shot text-to-image generation (2021)
21. Ren, P., Xiao, Y., Chang, X., et al.: A survey of deep active learning. CoRR abs/2009.00236 (2020)
22. Saunders, B., Camgoz, N.C., Bowden, R.: Adversarial training for multi-channel sign language production (2020)
23. Saunders, B., Camgoz, N.C., Bowden, R.: Everybody sign now: Translating spoken language to photo realistic sign language video (2020)
24. Simon, T., Joo, H., Matthews, I., et al.: Hand keypoint detection in single images using multiview bootstrapping. In: Cvpr (2017)
25. Stoll, S., Camgoz, N.C., Hadfield, S., et al.: Text2sign: Towards sign language production using neural machine translation and generative adversarial networks. Int. J. Comput. Vision 128(4), 891–908 (2020). https://doi.org/10.1007/s11263019-01281-2
26. Xiao, Q., Qin, M., Yin, Y.: Skeleton-based chinese sign language recognition and generation for bidirectional communication between deaf and hearing people. Neural Networks 125, 41–55 (2020). https://doi.org/https://doi.org/10.1016/j.neunet.2020.01.030

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Mejorando la Identificación de Marcas de Ganado Vacuno: Redes Siamesas en el Aprendizaje de Funciones de Distancia

Federico Stauber[1], Adrián Planas[2], Andrés Pascal[2]

[1,2] GIBD, Facultad Regional Concepción del Uruguay, Universidad Tecnológica Nacional
[1]fedestauber@gmail.com,
[2]{planasa, pascala}@frcu.utn.edu.ar

Abstract. Las Búsquedas por Similitud son importantes en diversas aplicaciones, incluyendo la identificación de marcas de ganado vacuno para el registro ganadero. Para calcular la similitud entre estas marcas, se utilizan funciones de distancia que miden dicha similitud en base a sus características, o en forma directa a partir de las imágenes correspondientes. En esta última década, las Redes Neuronales Profundas Convolucionales (CNN) han alcanzado muy buena performance en el procesamiento de imágenes. En este artículo se propone un método de preprocesamiento, aumentación de datos y modelos de CNN para aprender una función de distancia en un escenario de One-Shot learning utilizando una arquitectura de Redes Siamesas como mecanismo de entrenamiento.

Palabras Clave: Búsquedas por Similitud, Marcas de Ganado, Metric Learning, CNNs, Redes Siamesas, One-Shot Learning, Aumentación.

1. Introducción

Con la incorporación de nuevos tipos de datos como imágenes, audio, video y texto, los modelos de búsqueda tradicionales se vuelven insuficientes. La búsqueda por similitud emerge como una solución para encontrar objetos similares a un elemento de consulta específico en las bases de datos no tradicionales. Los Espacios Métricos [1] son un modelo que formaliza las búsquedas por similitud y permite el uso de métodos de acceso más eficientes.

Este artículo aborda el aprendizaje de funciones de distancia para la búsqueda de similitud de imágenes de marcas de ganado, una tarea esencial para la identificación y registro de nuevas marcas en el Registro Ganadero del Ministerio de Desarrollo Agrario de Buenos Aires. El registro de marcas de ganado está regulado por el Decreto Ley Nacional 22939 - SENASA 1983 [2], que establece en su Artículo III: *"No se admitirá el registro de un diseño de marca igual, ni uno que pueda confundirse con otro, dentro del ámbito territorial de la misma provincia o del territorio nacional. Se incluyen los que representen un diseño idéntico o similar y aquellos en los que uno de los diseños, al superponerse a otro, cubra todas sus partes"*.

Durante el registro de marcas de ganado, la búsqueda por similitud se realiza en forma manual debido a la falta de un sistema automatizado, afectando la eficiencia y eficacia del proceso. La dependencia de la habilidad visual de los empleados y la exhaustividad de la búsqueda se convierte en una limitación significativa.

Este problema presenta tres desafíos importantes para el aprendizaje automático: a) sólo existe una muestra de cada marca (One-Shot Learning), b) la cantidad de marcas está en crecimiento continuo y es muy costoso reentrenar el modelo cada vez que se agrega un nuevo elemento, y c) encontrar una función de distancia que mide correctamente la similitud entre dos imágenes no es una tarea trivial. En este artículo, se presentan modelos de CNN y técnicas de preprocesamiento, aumentación y entrenamiento para creación de una función de distancia que es eficaz en el cálculo del nivel similitud entre imágenes de marcas de ganado.

El resto de este documento está organizado de la siguiente manera: la Sección 2 presenta el trabajo relacionado; el proceso de aprendizaje de la función de distancia y los modelos utilizados se explican en detalle en la Sección 3. La Sección 4 describe los experimentos realizados y la Sección 5 los resultados obtenidos. Finalmente, en la Sección 6 se presentan las conclusiones y el trabajo futuro.

2. Trabajo Relacionado

Esta sección presenta el contexto del presente estudio. Se describen las técnicas claves para abordar el desafío de utilizar la CNN como función de distancia para la comparación por similitud de las marcas de ganado.

2.1 Recuperación de Imágenes Basada en Contenido

La Recuperación de Imágenes Basada en Contenido (CBIR) es un proceso que permite recuperar imágenes de una base de datos considerando ciertas características visuales, como la similitud en colores o formas. Los principales tipos de características utilizados son el color, la textura y la forma [3, 4]. El color se destaca como la característica visual más ampliamente utilizada, en gran parte debido a la facilidad de extraer datos cromáticos de las imágenes [5]. En contraste, obtener datos sobre la forma y la textura [6] es un proceso mucho más complejo y costoso.

Entre las soluciones más populares, los histogramas [7] ocupan un lugar destacado. Estos histogramas son computacionalmente eficientes, pero pueden ser insensibles a cambios en la posición de la cámara, carecen de información espacial y no manejan bien los cambios en el brillo general de la imagen [8].

La forma es una característica visual fundamental para describir el contenido de las imágenes, aunque puede verse afectada por defectos, ruido, oclusión y distorsiones arbitrarias, lo que dificulta su análisis. La forma puede ser descrita por diferentes aspectos [9] tales como: centro de gravedad [10], masa, media, dispersión, varianza, eje de menor inercia, rectángularidad y convexidad. Existen también enfoques más avanzados como el uso de descriptores invariantes: Momentos de Hu, Legendre o Zernike [11, 12, 13], que ofrecen resultados más confiables y precisos.

2.2 Búsquedas en Espacios Métricos

Los CBIR pueden ser generalizados y modelados mediante Espacios Métricos con el fin de lograr búsquedas eficientes. En [1], se muestra que el problema de búsqueda

de similitud puede ser formulado de la siguiente manera: dado un conjunto U de objetos y una función de distancia d definida entre ellos, el objetivo es recuperar todos los elementos cercanos a un objeto dado, utilizando d como criterio de similitud. Esta función d satisface las propiedades requeridas para ser una métrica:

(a) $\forall \in x\ U,\ d(x, x)=0$ *(reflexividad)*

(b) $\forall \in x, y\ U,\ d(x, y) \geq 0$ *(positividad)*

(c) $\forall \in x, y\ U,\ d(x, y)=d(y, x)$ *(simetría)*

(d) $\forall x, y, z \in U,\ d(x, z) \leq d(x, y)+d(y, z)$ *(desigualdad triangular)*

Un subconjunto finito X de U, denominado base de datos, se utiliza para realizar la búsqueda. Uno de los tipos de consulta relevante en las búsquedas por similitud es la consulta de los *k Vecinos Más Cercanos* o $NN_k(q)_d$, que recupera los k elementos de X más cercanos a q:

$$NN_k(q)_d = A = \{x \in X / \forall\ y\ \in (X\text{-}A),\ d(q, x) \leq d(q, y)\ \}\ \ y\ \ |A|=k$$

Dada una base de datos con n objetos, responder a estas consultas de forma trivial implica realizar n evaluaciones de distancia, lo cual puede resultar altamente costoso en aplicaciones prácticas. La relevancia de modelar estas consultas mediante espacios métricos se basa en la posibilidad de utilizar índices que aprovechan la propiedad de la desigualdad triangular para descartar elementos sin necesidad de compararlos directamente con la consulta. Esto mejora significativamente el proceso de búsqueda [1, 14, 15], convirtiéndolo en una solución eficiente en diversas aplicaciones.

2.3 Redes Neuronales Convolucionales (DCNN o CNN) y Redes Siamesas

La estructura de las Redes Neuronales Convolucionales (CNNs) fue propuesta inicialmente por Fukushima en 1988 [16] y durante los años '90, LeCun y otros investigadores [17] desarrollaron algoritmos de aprendizaje basados en el gradiente que mostraron resultados prometedores en la clasificación. Pero no fue hasta esta última década que las CNNs experimentaron mejoras significativas, destacándose en diversos problemas de reconocimiento de patrones.

Algunos ejemplos de las arquitecturas actuales de CNN son AlexNet [18], VGG Net [19] y NiN [20]. Otras alternativas más avanzadas y eficientes han sido propuestas, incluyendo DenseNet [21], GoogLeNet [22], y Residual Networks [23]. Los componentes básicos son casi los mismos para todas las arquitecturas, sin embargo, las diferencias topológicas producen distintos resultados tanto en la eficiencia en el entrenamiento como en la precisión.

Sin embargo, el concepto de similitud entre imágenes no se encuentra inherentemente presente en el algoritmo de aprendizaje de las Redes Neuronales Convolucionales (CNNs). No obstante, recientemente se han desarrollado arquitecturas como las Redes Neuronales Siamesas [24, 25] y Triplet Loss [26], que permiten capturar el concepto de similitud entre imágenes y han demostrado ser eficientes en el reconocimiento de rostros. Las Redes Siamesas consisten en un par de CNNs idénticas que pueden ser utilizadas para extraer vectores de características o para entrenar funciones de distancia directamente. Durante el entrenamiento, la red ajusta sus

parámetros para minimizar la distancia correspondiente a imágenes similares y maximizar la distancia correspondiente a imágenes diferentes.

El uso de Redes Siamesas/CNNs para la extracción de características orientadas a las Búsquedas por Similitud posee dos importantes problemas:

1. El primero es la escasez de instancias disponibles para el entrenamiento, problema conocido como One-Shot Learning o Few-Shots Learning [27, 28]. Esta situación imposibilita el entrenamiento directo de los modelos. La mejor estrategia hasta el momento para abordar este problema es el Transfer
   Learning/Fine Tuning [29, 30], sin embargo, esta técnica se ve limitada por la disponibilidad de bases de datos de características similares.
2. El segundo desafío es el de la generalización: cuando se entrena una CNN para extraer vectores característicos, el comportamiento del modelo suele estar fuertemente ligado a las clases con las cuales se realizó el entrenamiento. Por esta razón no logra adaptarse de manera efectiva a nuevas imágenes que se incorporen a la base de datos si no se la re-entrena. Una alternativa estudiada recientemente es el Aprendizaje Métrico (Metric Learning) [31], que propone aprender directamente una función de distancia correspondiente a los datos existentes en lugar de extraer características. Este es el enfoque utilizado en este estudio.

2.4 Aprendizaje de Funciones de Distancia (Metric Learning)

El aprendizaje métrico [32] se ocupa del aprendizaje de funciones de distancia que se ajustan a una determinada definición de similitud. En problemas de reconocimiento de patrones, la definición de similitud es específica a la tarea y el éxito del aprendizaje depende de la alineación del aprendizaje con la tarea en cuestión.

En [33] se realiza un análisis profundo de los métodos de aprendizaje de métricas y se da una visión general de las diversas técnicas y enfoques aplicados en el campo. En [34] se plantea un enfoque de aprendizaje métrico basado en márgenes. En otro trabajo relacionado, se describe una técnica de aprendizaje de métricas discriminativas aplicada al desafío de la verificación de rostros [35]. En [36] se proporciona una visión general del aprendizaje de métricas, cubriendo diferentes enfoques y aplicaciones y destacando los desafíos y tendencias en el campo.

2.5 Búsquedas por Similitud de Marcas de Ganado

En investigaciones previas se han usado varios enfoques sobre este tema. En [37] se emplearon histogramas de tangentes para cada punto de la imagen y se compararon utilizando el coeficiente de Pearson; otra propuesta consistió en el uso de histogramas generados midiendo distancias entre pares aleatorios de puntos de la imagen [38].

En un enfoque más prometedor, se obtuvieron mejores resultados en [39] utilizando Momentos de Hu y Legendre. En [40], se aplicó una CNN preentrenada para extraer características de las marcas de ganado, seguida de SVM para clasificación. Aunque alcanzaron una tasa de reconocimiento del 93%, persiste el desafío de agregar nuevas marcas a la base de datos, ya que éste método implica reentrenar el SVM cada vez que se incorporaba una nueva marca. Una solución más prometedora se presenta en [41],

donde se utiliza una técnica llamada ALR3, basada en histogramas de propiedades geométricas de las imágenes. Aunque muestra potencial, aún requiere una evaluación más extensa. En un estudio anterior [42], presentamos el uso de una CNN entrenada mediante Redes Siamesas y técnicas estándares de Aumentación de datos, junto con una técnica propia para mejorar la eficacia de las Búsquedas por Similitud de Marcas de Ganado.

3. Aprendizaje de la Función de Distancia

En esta sección se describe la base de datos y los modelos utilizados para el aprendizaje de funciones de distancia a partir de una base de datos compuesta por sólo una instancia de cada marca, utilizando Redes Siamesas, técnicas clásicas de Aumentación y el método de relieve presentado en [42].

3.1 Base de Datos de Marcas

El conjunto de datos utilizado consta de 203 marcas obtenidas del Registro Ganadero del Ministerio de Desarrollo Agrario de Buenos Aires. La Fig. 1 exhibe algunos ejemplos de estas marcas.

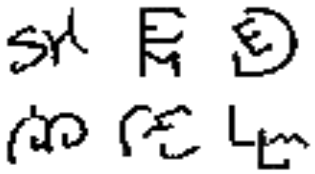


Fig. 1. Marcas de ganado pertenecientes a la base de datos.

3.2 Modelos utilizados

En este estudio se probaron doce modelos, de los cuales los tres mejores (Mod_1, Mod_2 y Mod_3) se muestran en este artículo. Los mismos consisten en CNNs de cinco (Mod_3) y seis (Mod_1 y Mod_2) capas convolucionales, todas con activación ReLU. Cada capa convolucional está intercalada con una capa de Batch-Normalization. A pesar de estas configuraciones en común, estos modelos se diferencian por las siguientes características:

- *Mod_1*: este modelo utiliza un tamaño de vector de salida de 128 elementos. Las tres primeras capas convolucionales emplean 32 kernels, mientras que las restantes utilizan 64. Se emplearon kernels de (3, 3) en cuatro capas y de (5, 5) en las dos restantes. Adicionalmente, se incorporan dos capas de Dropout para mitigar el overfitting. Finalmente, el modelo incluye una capa "flatten", que representa la salida que es utilizada por la red siamesa. Este modelo posee 193.664 parámetros entrenables.
- *Mod_2:* Este modelo utiliza un tamaño de vector de salida de 256 neuronas. Las cuatro primeras capas convolucionales emplean 32 kernels, mientras que las

restantes utilizan 64. Todos los kernels son de (3, 3). Adicionalmente, se incorporan una capa de MaxPooling2D y dos capas de Dropout. Luego se incluye una capa "flatten" antes de pasar a una capa densa de activación softmax, que representa la salida utilizada por la red siamesa. Este modelo cuenta con 664.080 parámetros entrenables.

- *Mod_3:* En este modelo, se utiliza la propiedad padding=’same’ en las capas de convolución para mantener las dimensiones de la imagen. Además, posee 5 capas convolucionales en lugar de 6, como los modelos anteriores. Las cuatro primeras capas convolucionales se intercalan entre 32 y 64 kernels, mientras que la restante es de 256. Los kernels son de (3, 3) para todas las capas convolucionales. Adicionalmente, incorpora una capa MaxPooling2D, dos capas de Dropout y una capa GlobalAveragePooling2D. Finalmente, el modelo incluye una capa "flatten" y una capa densa de 128 neuronas que representa la salida utilizada por la red siamesa. Contiene 237.280 parámetros entrenables.

A cada uno de los modelos presentados se lo utilizó para obtener representaciones de las dos imágenes a comparar, que luego se concatenan y pasan por dos capas densas con función de activación ReLU. Por último, se agrega una capa extra con la función Sigmoide que devuelve el valor final de similitud entre ambas imágenes.

Se emplea Binary Cross Entropy como función de pérdida y el optimizador Adam para el entrenamiento del modelo. Las imágenes utilizadas son de 28x28 píxeles.

3.3 Aumentación de Datos

En el ámbito de Búsquedas por Similitud, es común disponer de una sola muestra para cada objeto de la base de datos. Este hecho plantea un desafío significativo al emplear Redes Neuronales Convolucionales (CNNs) como método para determinar distancia de similitud, debido a las limitaciones mencionadas previamente. En el presente estudio, se solucionó este problema mediante la generación de 91.350 muestras adicionales (450 por cada marca) a partir de las 203 imágenes originales de la base de datos. Estas muestras se obtuvieron al azar, en principio mediante técnicas estándar de aumento de datos para imágenes en blanco y negro:

- Traslación: en direcciones este, oeste, norte y sur, hasta un 25% de la posición de la imagen original.
- Rotación: en sentido horario y antihorario, hasta un ángulo de 45 grados.
- Escalado: entre un 70% y un 130% del tamaño original.
- Cortes: hasta un 30% de la cantidad de píxeles de la imagen.
- Ruido Gaussiano: agregando hasta un 20% más de pixels en la imagen.

Y posteriormente se les agregó el procesamiento “Relieve”, propuesto en [42], que consiste en aplicar una expansión linealmente decreciente hacia los lados de las líneas que componen la imagen. Esta técnica produce mejoras significativas en la eficacia del entrenamiento, según nuestros experimentos. La Fig. 2 exhibe dos marcas (izquierda) después de su preprocesamiento y sus correspondientes relieves (derecha).

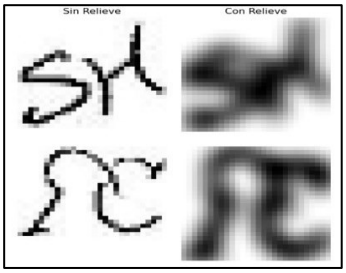


Fig. 2. Marcas y sus Relieves.

Los resultados experimentales demuestran que este mecanismo contribuye significativamente a que la Red Siamesa realice predicciones más robustas y estables en comparación con aquellas basadas exclusivamente en la aumentación estándar.

4. Experimentos Realizados

Para el entrenamiento de la Red Siamesa se seleccionaron al azar 406.000 pares de imágenes aumentadas, asegurando un equilibrio entre pares similares y no similares y garantizando la participación de cada marca en al menos 2.000 pares.

Se llevaron a cabo pruebas para evaluar la eficacia de los modelos mediante la función de distancia obtenida a partir de la Red Siamesa y los mecanismos de aumentación. Estas pruebas involucraron dos lotes de consultas diferentes, los cuales fueron:

- QD: 40 imágenes de consulta dibujadas a mano, las cuales se crearon copiando marcas existentes en la base de datos. Un total de 20 consultas se dibujaron en papel y posteriormente se digitalizaron, mientras que las otras 20 se dibujaron utilizando una aplicación de diseño gráfico.
- QN: para evaluar la capacidad de generalización de la función de distancia sin reentrenamiento, se incorporaron 10 nuevas imágenes a la base de datos y se generaron 10 consultas dibujadas a mano, una por cada nueva marca. La Red Siamesa no se reentrenó con las nuevas marcas. Este experimento tuvo como objetivo determinar si este método puede manejar nuevos elementos (nuevas "clases") de manera suficientemente efectiva sin requerir reentrenamiento.

Los modelos fueron entrenados en bloques de 5 épocas con 406.000 pares distintos de imágenes. En total se realizaron 150 épocas de entrenamiento alcanzando entre un 0,969 y un 0,995 de precisión. Se utilizó una PC con procesador Ryzen 7 5800x, 32 GB RAM y GPU GeForce GTX 1080 con 2560 núcleos CUDA.

Como métrica principal para evaluar la función de distancia, se empleó la tasa de aciertos (Hit Rate, expresada en porcentaje), dado que es el indicador más representativo para determinar si la solución propuesta es adecuada en el contexto del problema real planteado. Los resultados se compararon con el trabajo previo de extracción de características mediante redes siamesas y uso de la distancia euclidiana

como función de distancia, aplicado a la misma base de datos y lotes de consulta [42]. Este modelo anterior lo denominamos Mod_0 y lo usamos como base de comparación.

5. Resultados y Discusión

En la Tabla 1 se muestran los porcentajes de acierto en las búsquedas por similitud para los *k* vecinos más cercanos (NN_k), para k=1, 3 y 5. Como podemos observar, los modelos Mod_1 y Mod_3 superan al modelo anterior Mod_0 en las búsquedas de las imágenes dibujadas (QD) de las marcas con las que fueron entrenados. De estos dos, Mod_1 es el que obtiene mejores resultados. Sin embargo, ante el alta de nuevas marcas y búsquedas sin reentrenamiento (QN), Mod_3 resulta más consistente, con mayor poder de generalización, y supera significativamente al modelo anterior (90% para el NN_5 vs 60%).

Tabla 1. Porcentajes de Acierto de los distintos modelos.

Experimento	NN_1	NN_3	NN_5
Mod_0-QD	-	82,50	85,00
Mod_1-QD	**90,00**	**93,00**	**97,00**
Mod_2-QD	55,00	72,00	75,00
Mod_3-QD	80,00	**93,00**	95,00
Mod_0-QN	-	50,00	60,00
Mod_1-QN	10,00	50,00	80,00
Mod_2-QN	**30,00**	40,00	60,00
Mod_3-QN	20,00	**70,00**	**90,00**

Una característica importante de estas CNNs es que a pesar de ser profundas son relativamente pequeñas (Mod_3 posee solo 237.280 parámetros), lo que permite que su uso como función de distancia no sea excesivamente costoso.

A pesar de ello, hay que notar que estos modelos no cumplen con las propiedades de las distancias métricas. De hecho, la función obtenida no es reflexiva, ni simétrica y tampoco cumple la desigualdad triangular. La simetría se resuelve fácilmente haciendo d(x,y) = CNN(x,y) + CNN(y,x). Sin embargo, desde el punto de vista de los índices métricos, la propiedad que más importa es la desigualdad triangular, que es la que permite descartar elementos sin necesidad de compararlos con la consulta. Por esta razón, se realizó un experimento adicional para determinar en qué porcentaje la función de distancia satisface la desigualdad triangular para los elementos de la BD. Para ello se tomaron todas las ternas posibles y se verificó el cumplimiento de esta propiedad, alcanzando un 98,08% de éxito para Mod_3. Esto significa que sería posible utilizar índices métricos para acelerar la búsqueda, aun cuando en un porcentaje reducido de casos se podría estar descartando elementos erróneamente (índices métricos aproximados).

6. Conclusiones y Trabajo Futuro

En este artículo se presentaron modelos de CNN para el de Redes Siamesas, con Aumentación estándar de los datos y una técnica propia denominada Relieve para obtener una función distancia a utilizar en la búsqueda de imágenes de marcas de ganado ante una situación de One-Shot Learning. Como parte de este estudio se presentaron tres modelos de CNN, de los cuales uno de ellos alcanza niveles de acierto suficientemente altos como para ser implementado en un sistema real.

Las tareas actuales y las actividades previstas para el futuro próximo son las siguientes:

- Realizar experimentos sobre la BD completa (alrededor de 60.000 marcas).
- Probar el uso de índices métricos con Mod_3, para determinar su performance y el porcentaje real de elementos descartados erróneamente.
- Analizar estrategias para que la red generalice mejor, de tal manera de que sea robusta ante nuevas imágenes sin reentrenamiento.

Referencias

1. Chávez, Edgar, et al. Searching in metric spaces. ACM computing surveys (CSUR) 33.3: 273-321 (2001).
2. Decreto Ley Nacional 22939 SENASA. Servicio Nacional de Sanidad y Calidad Agroalimentaria. Título I - De las marcas y señales en general. Artículos 1 al 4. (1983).
3. Aslandogan, Y. Alp, and Clement T. Yu. Techniques and systems for image and video retrieval. IEEE transactions on Knowledge and Data Engineering 11.1: 56-63 (1999).
4. Smeulders, Arnold WM, et al. Content-based image retrieval at the end of the early years. IEEE Transactions on pattern analysis and machine intelligence 22.12: 1349-1380 (2000).
5. Irena, V., Rachev, B., Vassilakopoulos, M. Optimization of the algorithm for image retrieval by color features. CompSysTech. (2006).
6. Sarfraz, M., Ahmad, R. Content-based image retrieval using multiple shape descriptors. IEEE/ACS International Conference on Computer Systems and Applications (2007).
7. Pass, G., Ramin Z. Histogram refinement for content-based image retrieval. Proceedings Third IEEE Workshop on Applications of Computer Vision. WACV'96. IEEE, (1996).
8. Zhang, HongJiang, et al. Image retrieval based on color features: An evaluation study. Digital Image Storage and Archiving Systems. Vol. 2606. SPIE, (1995).
9. Zhang, Dengsheng, and Guojun Lu. Review of shape representation and description techniques. Pattern recognition 37.1: 1-19 (2004).
10. Traina, Agma JM, et al. Content-based image retrieval using approximate shape of objects. Proceedings. 17th IEEE Symposium on Computer-Based Medical Systems. IEEE, (2004).
11. Celebi, M. Emre, and Y. Alp Aslandogan. A comparative study of three moment-based shape descriptors. ITCC'05-Volume II. Vol. 1. IEEE, (2005).
12. Zhang, Dengsheng, and Guojun Lu. Content-based shape retrieval using different shape descriptors: A comparative study. ICME 2001. IEEE Computer Society, (2001).
13. Li, Shan, Moon-Chuen Lee, and Chi-Man Pun. Complex Zernike moments features for shape-based image retrieval. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans 39.1: 227-237 (2008).
14. Novak, David, Michal Batko, and Pavel Zezula. Metric index: An efficient and scalable solution for precise and approximate similarity search. Information Systems 36.4 (2011).

15. Britos, L., Printista, M., Reyes, N. DSACL+-tree: A dynamic data structure for similarity search in secondary memory. ICSSA. Springer, Berlin, Heidelberg (2012).
16. Fukushima, Kunihiko. Neocognitron: A hierarchical neural network capable of visual pattern recognition. Neural networks 1.2: 119-130 (1988).
17. LeCun, Yann, et al. Gradient-based learning applied to document recognition. Proceedings of the IEEE 86.11: 2278-2324 (1998).
18. Krizhevsky, Alex, Sutskever, Ilya and Geoffrey E. Hinton. ImageNet classification with deep convolutional neural networks. In Proceedings of the NIPS. (2012).
19. Simonyan, Karen, and Andrew Zisserman. Very deep convolutional networks for largescale image recognition. arXiv preprint arXiv:1409.1556 (2014).
20. Lin, Min, Qiang Chen, and Shuicheng Yan. Network in network. arXiv preprint arXiv:1312.4400 (2013).
21. Huang, Gao, et al. Densely connected convolutional networks. Proceedings of the IEEE conference on computer vision and pattern recognition. (2017).
22. Szegedy, Christian, et al. Inception-v4, Inception-Resnet and the Impact of Residual Connections on Learning. Thirty-first AAAI conference on artificial intelligence. (2017).
23. He, Kaiming, et al. Deep Residual Learning for Image Recognition. Proceedings of the IEEE conference on computer vision and pattern recognition. (2016).
24. Fierro, Atoany N., et al. Redes Convolucionales Siamesas y Tripletas para la Recuperación de Imágenes Similares en Contenido. Información tecnológica 30.6: 243-254 (2019).
25. Melekhov, Iaroslav, Juho Kannala, and Esa Rahtu. Siamese network features for image matching. 23rd international conference on pattern recognition (ICPR). IEEE, (2016).
26. Hoffer, Elad, and Nir Ailon. Deep metric learning using triplet network. International workshop on similarity-based pattern recognition. Springer, Cham, (2015).
27. Wang, Yaqing, et al. Generalizing from a few examples: A survey on few-shot learning. ACM computing surveys (csur) 53.3: 1-34 (2020).
28. Lake, Brenden, et al. One shot learning of simple visual concepts. Proceedings of the annual meeting of the cognitive science society. Vol. 33. No. 33. (2011).
29. Pan, Sinno Jialin, and Qiang Yang. A survey on transfer learning. IEEE Transactions on knowledge and data engineering 22.10: 1345-1359 (2009).
30. Storkey, Amos. When training and test sets are different: characterizing learning transfer. Dataset shift in machine learning 30: 3-28 (2009).
31. Schultz, Matthew, and Thorsten Joachims. Learning a distance metric from relative comparisons. Advances in neural information processing systems 16 (2003).
32. Brian Kulis. Metric Learning: A Survey. Foundations and Trends in Machine Learning (FTML), 5(4):287–364, 2013.
33. Yang, Liu, and Rong Jin. Distance metric learning: A comprehensive survey. Michigan State Universiy 2.2 (2006): 4.
34. Kilian Q. Weinberger and Lawrence K. Saul. Distance Metric Learning for Large Margin Nearest Neighbor Classification. J. Mach. Learn. Res. 10, 207–244. 2012.
35. Chopra, S., Hadsell, R., & LeCun, Y. (2005). Learning a similarity metric discriminatively, with application to face verification. In Proceedings, CVPR, IEEE. (2005).
36. Kulis, B. Metric Learning: A Survey. Now Foundations and Trends. (2013).
37. Sampallo, Guillermo M., et al. Reconocimiento de marcas de ganado. IX Congreso Argentino de Ciencias de la Computación (2003).
38. Sánchez Torres, Germán, and Manuel E. Rodríguez García. Medida de similaridad entre imágenes de marcas de ganado mediante distribuciones de forma. Revista Ingenierías Universidad de Medellín 13.25: 177-189 (2014).
39. Sanchez, G., and M. Rodriguez. Cattle marks recognition by hu and legendre invariant moments. ARPN Journal of Engineering and Applied Sciences 11.1: 607-614 (2016)
40. Silva, Carlos, et al. Cattle Brand Recognition using Convolutional Neural Network and Support Vector Machines. IEEE Latin America Transactions 15.2: 310-316 (2017).

41. Pascal, A., Michel, L., Romani, R. Zernike Moments vs ALR3 Applied to Similarity Searching of Cattle Brands. VI SABTIC (2018).
42. Pascal, A., Planas, A., Lederhos, F., Herrera, N. Extracción de Características utilizando Redes Siamesas y Aumentación aplicada a las Búsquedas por Similitud de Marcas de Ganado. CoNaIISI. Entre Ríos, Argentina (2022).

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Ciencia de Datos para el Desarrollo de un Modelo Predictivo de Heladas

María Isabel Masanet[1], Raúl Orcar Klenzi[1]

[1] Facultad de Ciencias Exactas, Físicas y Naturales,
Universidad Nacional de San Juan, San Juan, Argentina.
mimasanet@gmail.com, rauloscarklenzi@unsj-cuim.edu.ar

Resumen. En este trabajo se aplica Ciencia de Datos para el desarrollo de un modelo que predice el fenómeno meteorológico de la helada en la Provincia de San Juan, Argentina. A través del proceso sistemático que propone la Ciencia de Datos se ha logrado un modelo basado en una red neuronal recurrente de memoria de corto-largo plazo que, a partir de los valores de la temperatura y la humedad relativa censados cada diez minutos durante tres horas predice la temperatura hacia un horizonte de tres horas. La red fue entrenada, validada y testeada a partir de datos climáticos registrados por dos estaciones meteorológicas instaladas en la provincia de San Juan. Estos datos fueron preprocesados con un enfoque orientado hacia el análisis del fenómeno de la helada; además se aplicó la técnica de remuestreo SMOTEEN. Los resultados del modelo se analizaron con métricas de regresión y de clasificación.

Palabras clave: Ciencia de datos, Machine Learning, Red LSTM, Heladas.

1. Introducción

La Ciencia de Datos comúnmente se define como una metodología [1] mediante la cual se puede extraer conocimiento a partir de los datos, con el fin de utilizar este conocimiento para predecir eventos, comprender el pasado o presente, crear nuevos productos, entre otros usos. Tiene un objetivo ambicioso, pretendiendo generar opiniones basadas en los datos para que sean usadas en la toma de decisiones [2].

El proceso de la Ciencia de Datos es un proceso sistemático y disciplinado que involucra un conjunto de fases. En primer lugar, se debe entender el problema y plantear el objetivo de análisis de los datos. Luego, la etapa de exploración de los datos, para extraer características significativas que son usadas por herramientas de modelado y análisis, para finalmente obtener y presentar los resultados obtenidos [3]. En este punto, hay un ciclo de retroalimentación, el nuevo conocimiento disponible permite volver a la primera fase, a generar nuevas preguntas, nuevos problemas que se deben enmarcar y un nuevo proceso comienza.

Los proyectos de Ciencia de Datos suelen involucrar algún algoritmo de aprendizaje automático o Machine Learning (ML) [3]. El ML es un área de la

Inteligencia Artificial o Artificial Intelligence (AI) que busca aprender automáticamente relaciones y patrones significativos a partir de ejemplos y observaciones. Es decir, los algoritmos aprenden iterativamente de los datos de entrenamiento específicos del problema, lo que permite encontrar información oculta y patrones complejos sin ser programados explícitamente. Al aprender de cálculos anteriores y extraer regularidades de conjuntos de

datos masivos, ayuda a producir decisiones confiables y repetibles. Por esta razón, los algoritmos de ML se han aplicado con éxito en diversas áreas [4].

El clima es un proceso multidimensional, dinámico y caótico, y estas propiedades hacen que el pronóstico sea un gran desafío debido a la naturaleza no lineal de los datos meteorológicos [5]. Por esto el estudio del pronóstico del tiempo moderno implica una combinación de modelos informáticos complejos, observaciones in situ y el conocimiento de las tendencias y patrones climáticos mediante una metodología contemporánea avanzada, que ha llamado la atención de investigadores y científicos en diversos campos y disciplinas.

La helada es un fenómeno meteorológico localizado que puede ocasionar daño en diferentes niveles en los cultivos. Puede destruir toda la producción en cuestión de horas, provocando pérdidas de la cosecha de un año entero y comprometiendo ingresos del siguiente. Incluso si el daño no es visible inmediatamente después del evento, los efectos pueden surgir en el fin de temporada, reduciendo tanto la cantidad como la calidad de la cosecha [6].

Durante el año 2.020, en la provincia de Mendoza, el daño de las heladas tardías en el cultivo de vid fue estimado en un 24%, mientras que en el resto de las frutas fue del 84% [7]. En octubre de 2.021, se produjo una helada tardía que afectó a provincias de la Región de Cuyo. En algunas zonas de Mendoza la temperatura descendió hasta los -5°C. En San Juan la temperatura alcanzó durante la mañana los -1.8°C, ocasionando daños en los parrales [8].

En este trabajo se presenta el desarrollo de un modelo de predicción del fenómeno meteorológico de la helada a través de en una red neuronal recurrente de memoria a corto-largo plazo aplicando el proceso de Ciencia de Datos.

La organización del documento es la siguiente: la Sección 1 introduce los conceptos generales que explican el contexto y el propósito del trabajo; la Sección 2 presenta los trabajos relacionados a la predicción de heladas con ML; la Sección 3 describe el desarrollo del estudio y los resultados obtenidos. Finalmente, la Sección 4 presenta las conclusiones y sugiere posibles trabajos futuros.

2. Trabajos Relacionados

Diversos estudios se han llevado a cabo para el análisis y pronóstico de la temperatura. Los más recientes, aplican algoritmos de aprendizaje automático como redes neuronales de distinto tipo. En la mayoría de los casos, los datos son obtenidos de estaciones meteorológicas; aplicando técnicas de remuestreo cuando se presenta el inconveniente de escasez en la cantidad de datos reales.

Para predicciones hacia un horizonte a corto plazo, como es por hora y por día, se han usado los algoritmos de aprendizaje automático basados en modelos de redes neuronales, como multicapa perceptrón (MLP), memoria a largo-corto plazo (LSTM) y red neuronal de convolución (CNN), con datos meteorológicos de tres regiones de Corea del Sur, para el período de 2.009 a 2.018. Concluyendo que en la mayoría de los casos los datos de entrada por hora funcionaron mejor que los datos de entrada diarios. Además, en los resultados experimentales, dependiendo de la región de destino, el modelo CNN mostró el mejor rendimiento[9].

Un análisis comparativo de rendimiento de modelos para el pronóstico del clima con redes de memoria a largo-corto plazo (LSTM) y redes convolucionales temporales

(TCN) con enfoques clásicos (Regresión estándar, ARIMA, Random Forest, entre otros) presentan los autores de [10]; concluyendo que LSTM y el TCN producen un alto rendimiento y con errores más pequeños en comparación con los enfoques clásicos de aprendizaje automático y los enfoques de pronóstico estadístico.

Castañeda-Miranda y Castaño [11] desarrollaron un modelo autorregresivo con entrada externa (ARX) y otro con una red neuronal artificial (ANN) perceptrón multicapa, ambos capaces de pronosticar la temperatura interior de un invernadero. La conclusión fue que el modelo basado en ANN producía mejores resultados, con un nivel de confianza de 95%.

En Chile, con el objeto de predecir la temperatura mínima a partir de la información recopilada por diez estaciones meteorológicas desarrollaron una red neuronal artificial (ANN), con un número diferente de neuronas en la capa oculta (1, 5, 10, 15, 20, 25 y 30 neuronas). Los autores concluyeron que la ANN con 25 neuronas en la capa oculta fue el modelo que proporcionó los mejores resultados [5].

La técnica de sobremuestreo sintético de la clase minoritaria o Synthetic Minority Over-sampling Technique (SMOTE) fue aplicada en [6] para sobremuestrear los datos de cinco estaciones agrometeorológicas de la provincia de Mendoza, Argentina, a partir de los cuales trataron la predicción de heladas mediante un modelo desarrollado con redes Bayesianas y Random Forest. Al igual que la investigación de MöllerAcuña et all. [12], donde se generan datos a través de SMOTE.

En [13] el conjunto de datos obtenido de las estaciones meteorológicas es remuestreado con los métodos SMOTE, ADASYN, SMOTETomek y SMOTEEN. Cada conjunto de datos es entrada de un modelo basado en el algoritmo Random Forest que predice la ocurrencia o no ocurrencia de la helada. De la comparación de los resultados obtenidos concluyeron que SMOTEENN produce mejores resultados.

3. Desarrollo

El estudio se desarrolló a partir del proceso de Ciencia de Datos, que establece una secuencia de fases para la resolución de un problema. Esta secuencia no es estrictamente lineal, en cualquier momento del proceso es factible que sea necesario regresar a la fase anterior para hacer modificaciones. La Fig. 1 resume el flujo del trabajo realizado. Iniciando con el entendimiento del contexto del problema a través de información brindada por expertos. Se obtuvieron los datos desde dos estaciones meteorológicas, los cuales fueron sometidos a las tareas de análisis exploratorio. Luego prepararon en una estructura de ventana deslizante, adecuada para el procesamiento con algoritmos a usar. Posteriormente, en la fase de modelado se desarrolló una red neuronal de tipo LSTM. Finalmente, la evaluación del modelo se realizó a través de las métricas (recall, F1-Score y exactitud) y de la representación gráfica de las predicciones realizadas.



Fig. 1. - Proceso de Ciencia de Datos aplicado en la investigación. Fuente: Autora

3.1 Entendimiento del Contexto del Problema

Para el entendimiento del contexto del problema se solicitó información a expertos. Técnicamente, la palabra "helada" se refiere a la formación de cristales de hielo sobre las superficies, tanto por congelación del rocío como por un cambio de fase de vapor de agua a hielo [14]. Se produce cuando la superficie terrestre y el aire que se siente sobre ella alcanza una temperatura por debajo de los 0°C [15].

El pronóstico del fenómeno meteorológico de la helada a través de algoritmos de ML requiere de un conjunto de datos. Para este estudio se usaron los datos provistos por el Instituto de Automática (INAUT) de la Facultad de Ingeniería de la Universidad Nacional de San Juan, que correspondían a una estación meteorológica situada en el Establecimiento San Francisco S.A. (explotación privada), en la localidad de Cañada Honda, departamento Sarmiento, provincia de San Juan, Argentina. Otro conjunto de datos fue cedido por el Servicio de Agrometeorología de la Estación Experimental Agropecuaria San Juan dependiente del Instituto Nacional de Tecnología Agropecuaria (INTA) del departamento Pocito, donde se encuentra instalada otra estación meteorológica. Existiendo entre ellas una distancia de 37 km aproximadamente.

Ambas estaciones registran valores para las variables: temperatura exterior, humedad, velocidad y dirección del viento, precipitación, presión atmosférica, radiación solar, punto de rocío, evapotranspiración, índice de calor, índice de temperatura y humedad, entre otras. Realizan la medición y almacenamiento de los datos de las variables cada 10 minutos.

3.2 Análisis Exploratorio de los Datos

De la estación situada en INTA se consideraron los datos desde el año 2.016 hasta julio del año 2.021, y del Establecimiento San Francisco desde el mes de abril de 2.013 al año 2.018. De San Francisco se disponían un total de 299.671 registros, de los cuales 13.872 corresponden a heladas (aproximadamente el 4,63%). Para la estación INTA, el

total de registros es 301.899 que incluyen 3.854 casos de helada (aproximadamente el 1,28%).

Los datos registrados por las estaciones presentan una estructura tabular. La Fig. 2 muestra el preprocesamiento realizado a los datos.

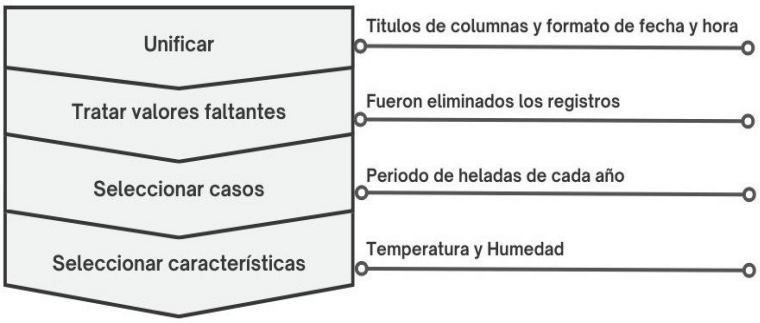


Fig. 2. Tareas de preprocesamiento de los datos

Unificar. Si bien ambas estaciones miden a igual intervalo de tiempo y las mismas variables, existieron diferencias en los conjuntos de datos que se debieron unificar, como nombre de columnas y formato de fecha y hora.

Tratar valores faltantes. Para el tratamiento de los valores faltantes existen distintas opciones. Una de ellas es eliminar los registros que poseen valores nulos. Otra es reemplazar los valores inexistentes por el valor medio de la columna, sin indicar que se hizo dicho reemplazo; o la opción anterior, pero agregando una columna al conjunto de datos, en dicha columna se identifica si el registro posee o no valores faltantes [16]. En este caso se optó por la primera opción.

Seleccionar casos. Por cada año se seleccionaron los datos correspondientes al periodo comprendido entre la fecha de la primera helada y la última helada. Las heladas son frecuentes entre las 00 horas y 8 horas, esto también se tuvo en cuenta para la selección de casos.

Seleccionar características. Para la selección de las características de interés se elaboró la matriz de correlación lineal con las variables meteorológicas: temperatura mínima, humedad relativa, punto de rocío, velocidad del viento, presión atmosférica y radiación solar.

La Tabla 1 muestra la cantidad de casos que se contiene el conjunto original y la cantidad de casos que resultaron luego de las tareas de selección, discriminando cantidad de heladas y no heladas.

Tabla 1. Cantidad total, de heladas y no heladas del conjunto original y el conjunto resultante de la selección de casos para cada estación.

Estación	Casos	Originales	Seleccionados
San Francisco	Total	299.671	82.975
	Heladas	13.872	13.814
	No Heladas	285.799	69.161
INTA	Total	301.899	49.973
	Heladas	3.854	3.830
	No Heladas	298.045	46.143

3.3 Preparación de los Datos

Los datos se estructuraron con la técnica de ventana deslizante, tomando los registros de las variables meteorológicas durante un periodo de 3 horas, este valor fue obtenido a través de un trabajo empírico desarrollado previamente.

El horizonte de predicción es de 3 horas, este el tiempo mínimo necesario para que el productor pueda desplegar las medidas de mitigación contra el fenómeno. La Fig. 3 representa la estructura de los datos.



Fig. 3. Ventana de entrada de 18 valores de cada variable meteorológica y salida con horizonte de 3 horas.

Tanto el conjunto de datos de entrenamiento como el de prueba deben ser representativos de los datos disponibles. En modelos que tratan de predecir el futuro a partir del pasado (por ejemplo, el clima de mañana, los movimientos de las acciones, etc.), no se deben mezclar aleatoriamente los datos antes de dividirlos, ya que al hacerlo se creará una fuga temporal [17]. La estrategia empleada para conformar estos conjuntos de datos fue usar los datos de la estación San Francisco para el entrenamiento y validación, y los datos de la estación INTA para test. Cada conjunto queda conformado por la cantidad de casos indicados en la Tabla 2.

Tabla 2. Cantidad de casos para cada conjunto de datos requerido para el modelado.

	Entrenamiento	Validación	Test
Total de casos	44.203	8.404	31.613
Casos de Heladas	9.838	2.843	3.760
Casos de No Heladas	34.365	5.561	27.853

En esta tarea de preparación de datos no fue posible la conformación de la ventana de 3 horas para algunos casos, debido a la falta de registro de datos ocasionada por problemas en la estación meteorológica.

Se puede observar claramente en la Tabla 2 que los conjuntos presentan un desbalance, superando notablemente la cantidad de casos de no heladas a los casos de heladas, siendo estos últimos el interés de la predicción. Este sesgo en el conjunto de datos de entrenamiento suele reflejarse en el rendimiento del modelo, siendo más preciso para los casos mayoritarios, que para los casos que son minoría.

En esta experimentación, el conjunto de datos de entrenamiento ha sido balanceado con la técnica de remuestreo SMOTEEN [18]. En un estudio previo concluyó que esta técnica arroja buenos resultados al aplicarla sobre estos datos para ser usados en el algoritmo de clasificación Random Forest [13]. Luego del remuestreo, la cantidad de casos que conforman el conjunto de entrenamiento y de validación usados para entrenar la red se detalla en la Tabla 3.

Tabla 3. Cantidad de casos para los conjuntos remuestreados.

	Entrenamiento	Validación
Total de casos	65.643	21.868
Casos de Heladas	34.000	10.810
Casos de No Heladas	31.643	11.058

3.4 Modelado

La arquitectura de red neuronal recurrente LSTM desarrollada consta de la capa de entrada con 18 valores de las variables temperatura y humedad relativa (36 valores en total); una capa oculta con función de activación RELU, por la facilidad que presenta esta función para el entrenamiento de las redes neuronales [19]; la función de pérdida (loss) el MSE por tratarse de un problema de regresión; y el algoritmo de optimización es Adam, con un valor para la tasa de aprendizaje de 8e-7. La capa de salida posee una neurona con función de activación lineal. La Fig. 4 representa la arquitectura de la red.

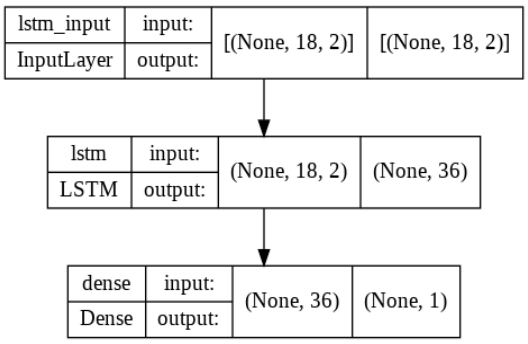


Fig. 4. Arquitectura de la red neuronal.

El entrenamiento de la red se realizó para 80 épocas (epochs). La curva de aprendizaje del modelo se muestra en la Fig. 5.

El modelo entrega como resultado el pronóstico de la temperatura a un horizonte de 3 horas. Luego, a partir del valor predicho para cada caso del conjunto de test, se realiza el etiquetado del caso con 0 o 1, según corresponda a una temperatura de helada o no helada respectivamente, transformando los resultados en un caso de clasificación binaria.

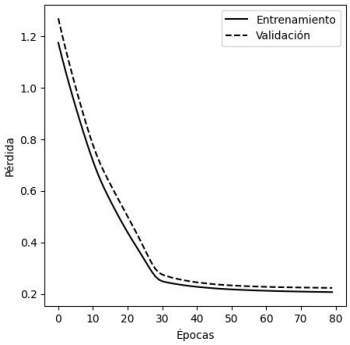


Fig. 5. Curva de aprendizaje del entrenamiento de la red neuronal.

3.5 Evaluación y Visualización

Para evaluar el modelo, los resultados arrojados por la red analizaron como un problema de regresión, a través del valor del MSE, RMSE y el coeficiente R^2. Luego del etiquetado y transformación a clasificación, se construyó la matriz de correlación y se analizó el valor recall y el F1-Score para los casos de heladas. Los resultados obtenidos por el modelo se presentan en la Tabla 4.

Tabla 4. Resultados de las métricas analizadas para el modelo desarrollado.

Métrica	Valor
R^2 entrenamiento	0,80
R^2 test	0,66
RMSE entrenamiento	2,52
RMSE test	2,60
Recall (Helada)	0.73
F1-Score (Helada)	0.69
Recall (No helada)	0.95
F1-Score (No helada)	0,96
Exactitud (Accuracy) en test	0.92

Las métricas de regresión (R^2 y RMSE) no presentan diferencias muy significativas, evidenciando que en el funcionamiento del modelo no hay sobreajuste. Se puede observar que el recall para los casos de heladas es de 0,73; lo que indica que el 73% de

las heladas son predichas correctamente. Y la exactitud del modelo es 0,92 contra 1 que es el valor ideal.

Además de analizar los valores de las métricas, se analizó la representación gráfica de la predicción realizada con el conjunto de test (la Fig. 6 muestra un recorte de la gráfica) donde se visualizó que la mayoría de los casos de heladas reales fueron predichos por el modelo.

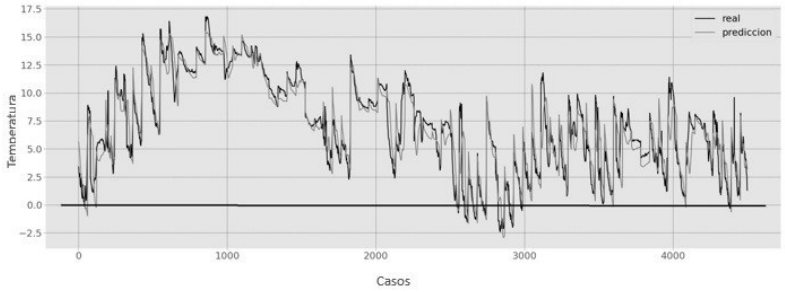


Fig. 6. Recorte de la representación de los valores reales del conjunto de test y sus correspondientes predicciones.

4. Conclusiones

Ante la necesidad de pronosticar el fenómeno meteorológico de la helada en una zona de la provincia de San Juan, Argentina, se llevó adelante este estudio que aplica el proceso de Ciencia de Datos para satisfacer el requerimiento.

Se logró desarrollar un modelo basado en una red neuronal LSTM que a partir de las variables meteorológicas temperatura y humedad relativa registradas a intervalos de 10 minutos durante un periodo de 3 horas, puede predecir la temperatura hacia un horizonte de 3 horas con una exactitud (accuracy) del 92% y la capacidad de pronosticar el 73% de casos de helada (sensbilidad o recall).

El proceso aplicado condujo a que los datos obtenidos desde las estaciones meteorológicas sean sometidos a una intensa tarea de preprocesamiento, para lo que ha planteado un flujo de trabajo que comienza con la unificación de los nombres de columnas y de formato de los datos, luego el tratamiento de valores faltantes, seguido de la selección de casos relevantes para el problema a tratar, y finalmente la identificación de las características de interés.

El método de remuestreo SMOTEEN aplicado a los datos estructurados en una ventana deslizante de 3 horas (18 valores) ha permitido generar conjuntos de datos equilibrados con los que fue entrenada la red neuronal.

Se puede concluir que el proceso de Ciencia de Datos ha permitido cumplir con el objetivo de desarrollar de forma sistemática y disciplinada un modelo de predicción del fenómeno de la helada con resultados aceptables.

En trabajos futuros incorporando datos desde otras estaciones meteorológicas y/o manteniendo los datos analizados, se pueden plantear mejoras al modelo, predicción de otras variables o considerar horizontes de predicción más lejanos.

Referencias

[1] L. Igual and S. Seguí, Introduction to Data Science. A Python Approach to Concepts,
[2] Techniques and Applications. Barcelona, España, 2017.
[3] S. Ozdemir, Principles of data science: learn the techniques and math you need to start making sense of your data. 2016.
[4] F. Cady, The Data Science Handbook, 1st. United States of America, 2017.
[5] C. Janiesch, P. Zschech, and K. Heinrich, "Machine learning and deep learning," Electron. Mark., vol. 31, no. 3, pp. 685–695, Sep. 2021, doi: 10.1007/S12525-02100475-2/TABLES/2.
[6] M. Fuentes, C. Campos, and S. García-Loyola, "Application of artificial neural networks to frost detection in central chile using the next day minimum air temperature forecast," Chil. J. Agric. Res., vol. 78, no. 3, pp. 327–338, Sep. 2018, doi: 10.4067/S0718-58392018000300327.
[7] L. Diedrichs, F. Bromberg, D. Dujovne, K. Brun-Laguna, and T. Watteyne, "Prediction of frost events using Bayesian networks and Random Forest," 2018.
[8] "Los daños por heladas tardías alcanzaron las 30 mil hectáreas de vid y 16 mil de frutales en Mendoza en 2020," 2021. https://www.infocampo.com.ar/los-danos-porheladas-tardias-alcanzaron-las-30-mil-hectareas-de-vid-y-16-mil-de-frutales-enmendoza-en-2020/ (accessed Apr. 14, 2022).
[9] "Alerta entre los productores de Cuyo por las heladas tardías - Revista InterNos," 2021.
[10] S. Lee, Y.-S. Lee, and Y. Son, "Forecasting Daily Temperatures with Different Time Interval Data Using Deep Neural Networks," Appl. Sci., vol. 10, no. 5, p. 1609, Feb. 2020, doi: 10.3390/app10051609.
[11] P. Hewage, M. Trovati, E. Pereira, and A. Behera, "Deep learning-based effective fine-grained weather forecasting model," Pattern Anal. Appl., vol. 24, no. 1, pp. 343– 366, Feb. 2021, doi: 10.1007/S10044-020-00898-1/FIGURES/12.
[12] Castañeda-Miranda and V. M. Castaño, "Smart frost control in greenhouses by neural networks models," Comput. Electron. Agric., vol. 137, pp. 102–114, May 2017, doi: 10.1016/j.compag.2017.03.024.
[13] P. Möller-Acuña, R. Ahumada-García, and J. Reyes-Suárez, "Predicción de Episodios de Heladas Basado en Información Agrometeorológica y Técnicas de Aprendizaje Automático," Dec. 2016, doi: 10.1109/ICA-ACCA.2016.7778386.
[14] M. I. Masanet, R. Klenzi, and F. Capraro, "Técnicas de balanceo de datos para predecir la ocurrencia del fenómeno meteorológico de la helada," Actas la XIX Reun. Trab. en Proces. la Inf. y Control. RPIC'2021, pp. 511–516, 2021, [Online].
[15] Available: https://drive.google.com/file/d/1byaIS-ssvJPSMHtKP9ahu8LQuoshyq6/view.
[16] R. L. Snyder, J. P. de Melo-Abreu, and J. M. Villar-Mir, "Protección contra las heladas: fundamentos, práctica y economía," Ser. FAO Sobre el Medioambiente y la Gestión los Recur. Nat., vol. 1, p. 257, 2010, Accessed: Mar. 30, 2020. [Online]. Available: http://www.fao.org.
[17] J. L. F. Yagüe, Iniciación a la meteorología y climatología. España, 2012.
[18] L. V. Gutiérrez, M. O. Ortega, M. I. Masanet, and F. De La Jara, "Técnicas de Análisis y Visualización de Minería de Datos para la Reducción de Dimensiones en Tablas de Datos," An. del Congr. Int. Ciencias la Comput. y Sist. Inf. 2019, 2019.
[19] F. Chollet, Deep Learning with Python, 1st ed. USA: Manning Publications Co., 2017.
[20] H. Kaur, H. S. Pannu, and A. K. Malhi, "A systematic review on imbalanced data challenges in machine learning: Applications and solutions," ACM Comput. Surv., vol. 52, no. 4, Aug. 2019, doi: 10.1145/3343440.
[21] C. Aggarwal, Neural networks and deep learning: a textbook, 1st ed. Cham, Switzerland: Springer Nature Switzerland AG, 2018.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

On the Precision Evaluation in Non-linear Sensor Network Design

José Hernández[1], Silvia Simón[1], Mercedes Carnero[1],
Gabriela Minetti[2], Carolina Salto[2,3],
[1] Facultad de Ingeniería, Universidad Nacional de Río Cuarto, Argentina
Ruta Nacional 8, Km 601
{jlh,ssimon, mcarnero}@ing.unrc.edu.ar
[2] LISI - Facultad de Ingeniería, Universidad Nacional de La Pampa, General Pico, Argentina
{minettig, saltoc}@ing.unlpam.edu.ar
[3]CONICET, Argentina

Abstract. The aim of this work is to provide a tool for evaluating the precision of process variable estimates in the context of the optimal design of a sensor network in chemical plants. One of the possible formulations for the optimal design of an instrumentation system for monitoring tasks is the solution of nonlinear optimization problems with constraints, where the objective function is the cost of the instrument and the constraints are the observability and global precision associated with a sensor placement. When a metaheuristic approach is used to solve this problem, a methodology for computing the constraints is needed to evaluate the quality of a proposed solution. A simulation technique has been selected to solve the precision associated with a set of measurements. The simulator requires a variable classification methodology and a data reconciliation function that consists of solving another non-linear optimization. The proposed strategies have been applied to a continuous stirred tank reactor, a nonlinear problem including flows, compositions, and temperatures related by mass and energy balances. Results demonstrating the performance of the proposed metaheuristics are presented.

Keywords: Sensor Network Design, optimization, Monte Carlo approach, variable estimates

1. Introduction

The availability of a reliable knowledge of the actual chemical plant state has a significant impact on its energetic efficiency and the fulfilment of environmental and safety regulations. This knowledge can be obtained installing an adequate sensor network at the design stage of the plant and applying data treatment procedures during its operation. Thus, data provided by sensors is transformed in consistent information to be used as input of monitoring, control and on line optimization methodologies. The number of sensors to be installed on the plant depends on the set of process variables that should be reliably known to execute the aforementioned procedures. The optimal selection of chemical plant instrumentation is defined as the Sensor Network Design Problem (SNDP). Because the estimation of the required variables can be attained by locating diverse types of sensors to measure different process variables, a huge combinatorial optimization problem arises even for small scale plants. Its solution

determines the set of process variables to be measured that optimizes the selected criteria and fulfills a set of constraints.

In general, the number of variables that are involved in these problems for a real work scenario is quite large and the formulation can be more or less complex depending on the performance criteria and the restrictions set used.

In this paper, the SNDP for process performance monitoring proposal is considered. Plant operation is represented by a nonlinear system of algebraic equations. The design minimizes the instrumentation cost, and also satisfies the observability of a required set of process variables and the precision of some of them. Precision is evaluated in terms of the standard deviation of variable estimates calculated using Monte Carlo approach.

The rest of this article is organized as follows. In Section 2 the SNDP is briefly reviewed, and the approach proposed for solving this problem is described. Section 3 presents one chemical process extracted from chemical engineering literature. An analysis of the results obtained using the proposed methodology is included in Section 4. Finally, the main conclusions and future lines of research are drawn in Section 5.

2. Problem Formulation

Let us assume R is a set of nonlinear algebraic equations that represents the operation of a process under steady state conditions

$$\mathbf{R}(\mathbf{z}) = \mathbf{R}(\mathbf{x}, \mathbf{u}) = \mathbf{0} \qquad (1)$$

where $\mathbf{z}$ is the n dimensional vector of process variables, and $\mathbf{x}$ and $\mathbf{u}$ are the vectors of measured and unmeasured variables, respectively. The optimal selection of instruments for process monitoring consists in determining the partition of vector $\mathbf{z}$ in vectors $\mathbf{x}$ and $\mathbf{u}$ that optimizes a given criterion and satisfies a set of constraints. In general, restrictions are imposed on the degree of estimability of key process variables and also on the quality of their estimates. One particular selection problem obtains the minimum cost sensor network that satisfies a degree of estimability greater or equal to 1 and estimates precision constraints for a given set of required variables. This problem is stated by the following equation

$$\begin{aligned} \min \quad & \mathbf{c}^{\mathrm{T}}\,\mathbf{q} && (2) \\ \text{s.t.} \quad & \\ & E_l(\mathbf{q}) \geq 1 && \forall\, l \in S_E \\ & \hat{\sigma}_k(\mathbf{q}) \leq \sigma_k^*(\mathbf{q}) && \forall\, k \in S_\sigma \\ & \mathbf{q} \in \{0,1\}^n \end{aligned}$$

where $\mathbf{q}$ is an n-dimensional vector of binary variables such that $q_i = 1$ if variable i is measured, and $q_i = 0$ otherwise, $\mathbf{c}^{\mathrm{T}}$ is the cost vector; E_l stands for the degree of estimability of the l-th variable included in S_E and $\hat{\sigma}_k$ is the estimate standard deviation of the k-th variable contained in S_σ after a data reconciliation procedure is applied.

Furthermore, S_σ and S_E are the set of key process variables with requirements on estimability and precision, respectively.

In this formulation, measurements are subject to non correlated random errors; there is only one potential measuring device for each variable, and there are no restrictions for the localization of instruments.

Furthermore, the feasibility of the constraints can be checked by executing variable classification and data reconciliation procedures, [1].

Tree-search algorithms, nonlinear mixed-integer programming techniques, and heuristic techniques have been used to solve the SNDP, [2] [3] [4] [5]. When a metaheuristic approach is used to solve this problem, a methodology for computing the constraints is needed to evaluate the quality of a proposed solution.

The precision constraint associated with the estimation of a required variable can be evaluated explicitly in linear systems, but not in nonlinear ones. One approach is to perform a linearization of the balance equations governing the plant model around a known operating point. The precision of the required variables can then be calculated using a linear data reconciliation methodology. In this case, it is possible to have an explicit expression for the deviation calculation with expressions that involve calculating the inverse of matrices that may have high condition numbers [6].

A second option is to implement a Monte Carlo method, which has the advantage of flexibility and simplicity, but whose accuracy depends on the size of the sample with which the simulation is performed.

The objective of this paper is to propose a methodology based on the second option for precision estimation that can be implemented in optimal sensor placement methodology.

The feasibility of the solutions determined by a simulation procedure involves the resolution of non-linear optimization problems at a second level. The optimization strategy is dedicated to solve the following non-linear data reconciliation problem

$$\underset{\mathbf{x},\,\mathbf{u}}{Min}\ (\mathbf{y}-\mathbf{x})^T\,\Sigma^{-1}(\mathbf{y}-\mathbf{x}) \qquad (3)$$

subjet to

$$\mathbf{f}(\mathbf{x},\mathbf{u}) = \mathbf{0}$$

$$\mathbf{g}(\mathbf{x},\mathbf{u}) \le \mathbf{0}$$

where **f** and **g** are the vectors of equality and inequality constraints, respectively, **Σ** stands for the variance-covariance matrix, and **y** represents the measurement vector. For a given set of measured variables, the estimates vector attained solving problem (3), $\hat{\mathbf{z}}$, is a vector of random variables of certain joint probability distribution. The population standard deviation vector of $\hat{\mathbf{z}}$ can be estimated by a simulation procedure that generates a set of **y** vectors, obtains the corresponding $\hat{\mathbf{z}}$ vector for each **y** vector, and then calculates the sample standard deviation vector of $\hat{\mathbf{z}}$. The application of a variable classification technique and the data reconciliation methodology allows evaluating the estimate precision of each variable included in S_σ as it will be explained next.

Each solution must be evaluated to determine the fitness of the proposal. An approach based on penalty functions is used to handle the constraints of the optimization problem given by Eq. (1).

The working hypothesis is that the critical stage, in terms of computational effort, is the feasibility check of a solution associated with the precision calculation and, consequently the scaling of the SNDP for nonlinear systems needs the implementation of an additional strategy to manage the verification of precision constraints in a more bounded way. The following is an outline of the procedure for the calculation of fitness (Fig. 1).

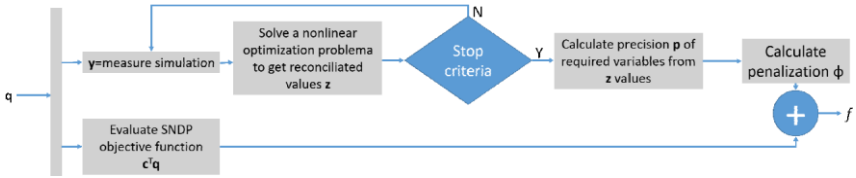


Fig. 1. Fitness calculation scheme with estimability constraints

2.1 Proposed solution approach: Estimates Precision of required variables

This subsection addresses the estimate variance calculation for a required variable at the instrumentation design stage of a non-linear process. For a potential solution **q**, this calculation depends on the results of the variable classification procedure, therefore let us revise some concepts related to this technique.

Process variables can be measured or unmeasured. A measured variable is classified as redundant if the measurement value can be adjusted by a data reconciliation procedure. If it is not the case, the measured variable is non-redundant. Regarding unmeasured variables, they are categorized as observable or unobservable. The first ones can be calculated using the measurements and the process model, while the second ones cannot be estimated.

Estimate variance calculation is straightforward for non-redundant measured variables. In this case the estimate variance is equal to the measurement variance, because the measurement value cannot be adjusted. Regarding non-observable unmeasured variables, their estimate variances are also calculated directly. They are infinite because those variables cannot be calculated.

The data reconciliation procedure allows estimating the redundant measured variable using **y** and the process model. In general form, the reconciled or adjusted value of the *i*-th redundant measured variable can be expressed as shown in Eq. 4.

$$\hat{X}_i = h_i(Y_1, Y_2, \ldots, Y_m) \qquad (4)$$

where $(Y_1, Y_2, \ldots, Y_m)$ represents the random vector of measurements associated to a particular SN configuration. Given a realization of that vector, $\hat{X}_i$ is also a random variable. For specific functions h_i, the probability density function of $\hat{X}_i$ can be determined using variable transformation techniques. Therefore, its parameters (mean

value and variance of $\hat{X}_i$) are known. A particular case arises if plant model is linear because analytical expressions of the reconciled variable value and its standard deviation is available.

In contrast, for non-linear process models the point estimate of the redundant measured variable is obtained by applying a numerical method, this is a non-linear optimization strategy. In this work, the weighted least square technique subject to non-linear process constraints is applied to obtain the point estimate, and a methodology is devised to calculate estimate variance from point estimates.

The treatment of unmeasured observable variables, U_j is similar to the one proposed for the redundant measured ones, except that in general they are expressed as a function of r reconciled variable values as Eq. 5 shows.

$$U_j = g_j (\hat{X}_1, \hat{X}_2, \dots , \hat{X}_r) \quad (5)$$

Variable U_j is also a random variable. If function g_j satisfies specific conditions, the probability density function of U_j can be determined using variable transformation techniques to known its parameters (mean value and variance of U_j).

In the rest of the subsection, a Monte Carlo Simulation methodology to calculate the estimate variance of redundant measured variables and observable unmeasured ones from point estimates is analysed. Furthermore, it is considered from this point below that the expected value of $\hat{X}_i$ is equal to the nominal value of the variable used at the design stage of the plant, measurements are independent and the probability density function of each measurement is normal (nominal value, sensor standard deviation).

2.2 Monte Carlo Simulation

The Monte Carlo simulation, MC, is a stochastic numerical procedure widely used in Engineering to simulate the behavior of a phenomenon or a real process that involves random variables such that their probability density functions are known. Its implementation is quite simple but it may require huge computational resources.

The scheme proposed to calculate the precision of measured variables for a solution vector **q** is presented in Alg. 1.

```
:input solution vector q
required process variables vector req
:output
vector of standard deviations of required variables std
N=0
/* run variable classification procedure */
cla=variable_classification(q,req)
if all req variables are measured and non redundant
     std= vector of standard deviation of instruments
elseif all req variables are measured or observable
      while not(stop_criteria)
         yob=sample of the random vector of measurements
```

```
        z0=Data_reconciliation(yob /*Get estimation*/
        N=N+1
     end_while
     std=Calculate_deviation_of_sample_reconciled_values
else
     std=Inf      /*no feasible solution*/
end_if
```

Algorithm 1. Pseudocode of MC

The sampling of the random measurement vector is performed considering the probability density function of each measurement. The stopping criterion used ensure that the estimated deviation value is bounded, within a small tolerance range ε, for a preset number of iterations, n_ε.

3. Results

This section presents a nonlinear application example. The case study is a continuous stirred tank reactor (CSTR) [7] whose model comprises 13 variables (total flow rates, compositions, and temperatures), 5 mass and energy balances and the reaction rate equation given by Eq. 6 to 11.

$$\frac{dC_A}{dt} = \frac{F_i}{V}(C_{Ai} - C_A) - r_A \quad (6)$$

$$\frac{dV}{dt} = Fi - F \quad (7)$$

$$\frac{dT}{dt} = \frac{F_i}{V}(T_i - T) + \frac{r_A(-\Delta H)}{\rho C_p} - \frac{UA(T - T_C)}{V \rho C_p} \quad (8)$$

$$\frac{dT_C}{dt} = \frac{F_C}{V}(T_{Ci} - T_C) + \frac{UA(T - T_C)}{V_j \rho_j C_{pj}} \quad (9)$$

$$\frac{dn}{dt} = r_A V - F_{vg} \quad (10)$$

$$r_A = C_d C_A k_0 \exp(-E/RT) \quad (11)$$

The process flowsheet is shown in Fig. 2.

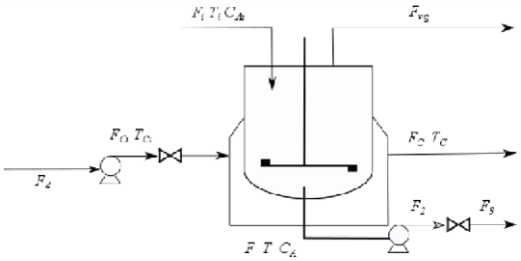


Fig. 2. Process Flowsheet – CSTR

To evaluate the performance of the proposed methodology, a design problem which comprises three required variables is considered; these are C_A, T and F. The precision constraints are σ_3^* =2.227 10^{-3} , σ_4^* =5.700 and σ_{10}^* =3.800 10^{-1}.

Table 1. True variable values - CSTR

#	Variable	*Nominal value*	*Cost of instrument*	#	Variable	*Nominal value*	*Cost of instrument*
1	F_i	40.04	100	8	T_{ci}	530.64	35
2	C_{Ai}	0.5	270	9	F_{vg}	10.61	85
3	C_A	0.23	300	10	F	40.04	90
4	T	600.03	50	11	F_2	40.04	95
5	T_i	529.01	55	12	F_3	40.04	80
6	T_c	590.61	60	13	F_4	56.69	82
7	F_c	56.69	105				

Results obtained for three particular solutions, denoted as $\mathbf{q}^1$, $\mathbf{q}^2$ and $\mathbf{q}^3$, are examined next. All the process variables are measured in $\mathbf{q}^1$. Therefore, all variables are redundant and the reconciliation procedure is called to calculate the precision of the required variables. The same happens for $\mathbf{q}^2$, where the proposal is to measure CAi, FVg, F3 variables. In this case, the required variables are unmeasured and observable, then they are evaluated in terms of the measurements. Regarding q3, it corresponds to the solution of the SNDP for the CSTR, that is q3= q*, In this case the measured variables are CAi, CA, Tci y F3. Figures 3, 4 and 5 display the reconciliated values of the required variables and the deviation of the sample mean of these values with respect to the true values for an increasing number of simulation trials.

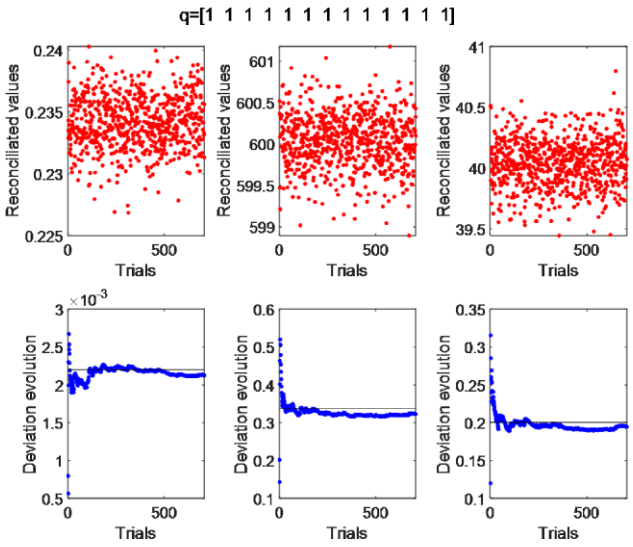


Fig. 3. Reconciled values and deviation of the sample mean for q1

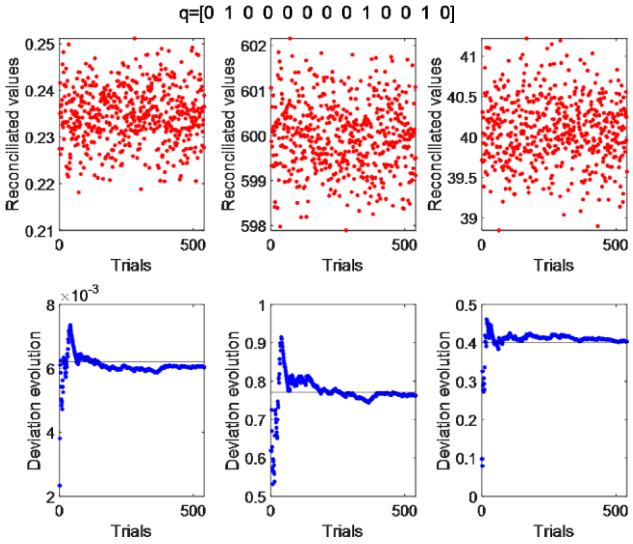


Fig. 4. Reconciled values and deviation of the sample mean for q1

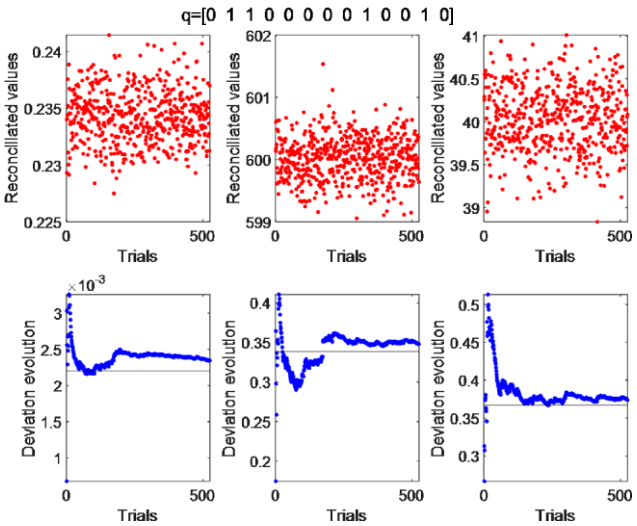


Fig. 5. Reconciled values and deviation of the sample mean for q*

The convergence of the sample mean of the reconciled values to the true ones is attained faster than the stabilization of the sample standard deviation around a certain value. Furthermore, the number of MC simulation trials required to achieve the convergence of the sample standard deviation depends on the analyzed solution. The simulation study performed allows inferring that MC method requires N=1000 trials, in average, to provide a reliable estimate of the precision for a required variable. Therefore, an upper bound of the total computational effort necessary to verify the feasibility of precision constraints for a potential solution is N* # required variables. For the CSTR problem, the maximum number of calls to the data reconciliation function for each solution to be evaluated is 3000. To solve problems of larger size, MC utilization should be revised.

4. Conclusions

In this work, a strategy has been developed for the precision evaluation associated with the estimation of a required variable in the optimal instrumentation schemes resolution framework for chemical plants. It is assumed that the plant model is represented by a set of global mass balances, by component and energy balances in steady state, which implies that the precision constraint evaluation involves a simulation method together with the solution of a nonlinear data reconciliation problem. All these features bring more generality to the optimal sensor network design problem and also more complexity to solve it. The proposed Monte Carlo simulation method exhibits good results, although the computational complexity may be very high for problems with a large number of required variables. Nevertheless, the method allows to be used as a validation tool even in scenarios where the characteristics of the network

require the implementation of strategies to rationalize the effort to obtain solutions in limited time.

5. Acknowledgments

The authors acknowledge the support of Universidad Nacional de La Pampa (Project FI-CD-107/20), Universidad Nacional de Río Cuarto, and the Incentive Program from MINCyT, Argentina. The last two authors are grateful for the support of the HUMAN-CENTERED SMART MOBILITY (HUMOVE) project, PID2020-116727RB-I00, Spain. The last author is also funded by CONICET, Argentina.

References

1. Romagnoli, J.; Sánchez, M. Data Processing and Reconciliation for Chemical Process Operations; Academic Press: San Diego, 2000).
2. Carnero, M., Hernández, J.L., Sánchez, M. Optimal sensor location in chemical plants using the estimation of distribution algorithms. Industrial & Engineering Chemistry Research, vol. 57(36), pag 12149–12164, (2018).
3. Zhang, Jin and Donald J. Chmielewski, 2017 "Profit-based Sensor Network Design using the Generalized Benders Decomposition." American Control Conference, (ACC) 3894-3899.
4. Hernandez J., Salto C., Minetti G., Carnero M., Sanchez M.C., Hybrid Simulated Annealing for Optimal Cost Instrumentation in Chemical Plants, Chemical Engineering Transactions, 74, 709-714 (2019)
5. He, Y.J., Ma, Z.F. 2014, "Optimal Design of Linear Sensor Networks for Process Plants: A Multi-Objective Ant Colony Optimization Approach". Chemometrics and Intelligent Laboratory Systems, 135, 37-47, 2014.
6. Nguyen, DQ, Bagajewicz, M. Design of Nonlinear Sensor Networks for Process Plants. Ind. Eng. Chem. Res. 2008. .47, 5529–5542.
7. Bhushan, M.; Rengaswamy, R. Design of Sensor Network Based on the Signed Directed Graph of the Process for Efficient Fault Diagnosis. Ind. Eng. Chem. Res., 39, 999, (2000).

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Variantes Dispersivas para el Operador de Mutación en Algoritmos Genéticos con Cromosmas Binarios

Ana Laura Maffei y Tomás Tetzlaff

Universidad Nacional de Lujan, Departamento de Ciencias Básicas,
Luján, Buenos Aires, Argentina

Universidad Nacional de General Sarmiento,
Instituto de Ciencias Los Polvorines, Buenos Aires, Argentina

Resumen Para algoritmos genéticos con cromosomas cuyos genes pueden tomar valores 0 o 1, analizamos la distribución del número de mutaciones por cromosoma con distintos operadores de mutación. Consideramos en primer lugar la operación clásica de recorrer las variables individuales binarias del cromosma y para cada una cambiar su valor independientemente con igual probabilidad μ, que resulta en una distribución binomial para el número de mutaciones por cromosoma. Luego definimos una familia de distribuciones de un parámetro que mantienen la media del operador binomial pero alteran la varianza con el fin de modificar el balance entre exploración y explotación en el espacio de búsqueda. Hallamos propiedades de dispersión de estos operadores y analizamos su desempeño en ensayos computacionales.

Keywords: Algoritmo Genético, Mutación, Dispersión de Cromosomas.

1. Introducción

El procedimiento usual para la mutación de un cromosoma en algoritmos genéticos consiste en recorrer uno por uno los genes del cromosoma y mutarlos independientemente con una probabilidad constante μ. Una estrategia para resolver el problema del estancamiento de varios o todos los elementos de la población de cromosomas en un máximo local (no global) consiste en medir la dispersión de la población de cromosmas y si se encuentran muchos repetidos o alguna otra manifestación de una dispersión muy baja [4], aumentar la probabilidad de mutación con el fin de salir de una zona favorable pero no óptima [3]. Esta estrategia tiene la desventaja de tener que decidir cambios en la probabilidad de mutación de una manera que puede ser automática pero arbitraria. Además, el aumento del número esperado de mutaciones puede conseguir una mejor exploración del espacio, pero dificultar la explotación de vecindades pequeñas una vez que una nueva zona favorable es encontrada. En el presente trabajo estudiamos la posibilidad de mantener el mismo operador de mutación durante toda la corrida del algoritmo sin necesidad de cambiar en momentos que podrían ser

arbitrarios. Para ello analizamos una familia de operadores que mantienen el valor esperado del número de mutaciones, pero con distintas distribuciones de ese número aleatorio. Presentamos propiedades probabilísticas de estas distribuciones y analizamos el desempeño de los correspondientes operadores en ensayos computacionales con instancias aleatorias del problema de la mochila mediante algoritmos gen éticos sin cruce con el fin de priorizar la relevancia de la mutación.

2. Distribución Binomial del Número de Mutaciones y Alternativas

La operación clásica de mutación es la de recorrer los genes de cada cromosoma y mutar cada gen, a 1 si es 0 y a 0 si es 1, independientemente con probabilidad constante μ. Si la longitud de un cromosoma es l, esta operación representa l ensayos independientes de Bernoulli, cada uno con probabilidad de éxito μ, es decir que el número de mutaciones en un cromosoma sigue una distribución binomial de parámetros l y μ. Si bien hay variadas propuestas sobre el valor conveniente μ [1] el valor de μ usualmente utilizado es l^{-1} [2], esto es, la inversa de la longitud del cromosoma. Por lo tanto, este procedimiento es equivalente a seguir los siguientes pasos.

(1) Obtener el número de mutaciones para el cromosoma de una distribución binomial, $Bi\left(l, l^{-1}\right)$.

(2) Tomar una muestra aleatoria sin reposición de los l genes en la que cada gen tiene igual probabilidad de formar parte de la muestra.

(3) Mutar los genes de la muestra.

Dada la conveniencia de buscar un óptimo global en una zona donde los cromosomas tienen alto fitness, pero procurando no quedar atrapados en un máximo local, los algoritmos gen éticos buscan un adecuado balance entre explotación y exploración. La primera es la búsqueda en una zona con valores altos de fitness comparados con lo obtenido hasta el presente por el algoritmo. La segunda es la incursión en zonas del espacio de búsqueda más alejadas e inciertas pero que podrían contener un óptimo global. En este trabajo modificamos el paso (1) cambiando la distribución binomial por otra que podría mejorar la capacidad de explotación y exploración. Para ello sustituimos la distribución binomial por otra con la misma media (valor esperado del número de mutaciones) la cual es igual al número de ensayos de Bernoulli multiplicado por la probabilidad de éxito en cada ensayo, lo cual resulta ser $ll^{-1} = 1$. Buscamos que la nueva distribución tenga una alta probabilidad de ir mutando los genes de a uno para una adecuada explotación del entorno cercano pero que cada tanto realice un cambio completo en algún cromosoma mutando todos sus genes para ubicarlo en las antípodas del espacio de búsqueda, es decir a distancia de Hamming máxima (l) del cromosoma original. Esto se consigue sin modificar la esperanza de la cantidad de mutaciones.

3. Familia E_l de Distribuciones con Media 1

Para una longitud de cromosoma $l > 1$, sea

$$D_l = \left\{ (p_0, p_1, ..., p_l) \in [0,1]^l : \sum_{i=0}^{l} p_i = 1, \sum_{i=0}^{l} i p_i = 1 \right\}.$$

Es decir, D es el conjunto de distribuciones de probabilidad sobre $\{0, 1, ..., l\}$ con media 1. Definimos $E_l \subset D_l$ como el conjunto de distribuciones de D_l tales que $p_i = 0$ si $i \notin \{0, 1, l\}$. Veamos que todo elemento de E_l queda determinado por un solo parámetro que denominamos p y que es la probabilidad p_1 de que el cromosoma tenga una sola mutación. En efecto, si $p = p_1$, para tener esperanza 1 debemos tener $0p_0 + 1p + lp_l = 1$

y por lo tanto $p_l = (1-p)/l$, mientras que la suma de probabilidades igual a 1 determina que $p_0 = 1 - p_1 - p_l = 1 - p - (1-p)/l = (1-p)(1-l^{-1})$

Podemos calcular la varianza de un elemento de $e_p \in E_l$ en función de $p = p_1$ comenzando con el cálculo del segundo momento el cual es $0^2 p_0 + 1^2 p + l^2 p_l = p + l^2 (1-p)/l = p + l - lp = p(1-l) + l$. Restando el cuadrado de la esperanza obtenemos $V e_p = p(1-l) + l - 1^2 = -p(l-1) + l - 1 = (1-p)(l-1)$.

Nos interesa verificar que las varianzas de los elementos de E_l recorren todos los valores posibles de varianza para distribuciones de probabilidad con media 1 sobre $\{0, 1, ..., l\}$ para lo cual utilizamos las siguientes propiedades.

Propiedad 1 *La varianza de la distribución de e_p en la que $p = 1$ es la mínima sobre todas las varianzas de los elementos de D_l.*

Demostración 1 *Por ser una distribución concentrada en un punto, la varianza es en este caso 0 y no puede existir una distribución con varianza menor.*

Propiedad 2 *La varianza $V e_0$ de la distribución de $e_0 \in E_l$ en la que $p = 0$ es la máxima sobre todas las varianzas de los elementos de D_l y toma valor $V e_0 = l - 1$.*

Demostración 2 *Como D_l es un conjunto cerrado y acotado y la varianza es una función continua, existe una distribución $d \in D_l$ en la que se alcanza varianza máxima. Supongamos que $d \notin E_l$ y llegaremos a un absurdo. Sean $\{q_0, q_1, ..., q_l\}$ las probabilidades que definen a d. Si $d \notin E_l$ existe j tal que $2 \leq j \leq l-1$ con $q_j > 0$. Vamos a construir una distribución $d_e \in D_l$ tal que supera en varianza a d, es decir $V\tilde{d} > Vd$, absurdo. Definimos las probabilidades $\{\tilde{q}_0, \tilde{q}_1, ..., \tilde{q}_l\}$ de $\tilde{d}$ a continuación.*

$$\tilde{q}_i = q_i \ \ si \ i \notin \{0, j, l\}$$

$$\tilde{q}_0 = q_0 + \left(1 - \frac{j}{l}\right) q_j$$

$$\tilde{q}_j = 0$$

$$\tilde{q}_l = q_l + \frac{j}{l} q_j$$

Es decir que la masa sobre j se reparte (convenientemente como veremos) entre 0 y l. Es fácil verificar que $\sum_{i=0}^{l} \tilde{q}_i = 1$. Además

$$\begin{aligned}\sum_{i=0}^{l} i\tilde{q}_i &= 0\left(q_0 + \left(1 - \frac{j}{l}\right) q_j\right) + 1q_1 + ... + j0 + ... + l\left(q_l + \frac{j}{l} q_j\right) \\ &= 0 + 1q_1 + ... + j0 + ... + lq_l + jq_j \\ &= \sum_{i=0}^{l} iq_i = 1\end{aligned}$$

por lo cual $\tilde{d} \in D_l$. Ahora, como la varianza es igual al segundo momento menos el cuadrado de la esperanza y esta es 1 para ambas distribuciones, basta ver que el segundo momento de $\tilde{d}$ es mayor que el de d, lo cual se cumple porque

$$\begin{aligned}\sum_{i=0}^{l} i^2\widetilde{q_i} &= 0 + 1^2 q_1 + \ldots + j^2 0 + \ldots + l^2 q_l + l^2\frac{j}{l}q_j \\ &= \sum_{i=0}^{l} i^2 q_i - j^2 q_j + l^2\frac{j}{l}q_j \\ &= \sum_{i=0}^{l} i^2 q_i + q_j j\,(l-j) \\ &> \sum_{i=0}^{l} i^2 q_i\end{aligned}$$

como queríamos probar. Finalmente el valor de V e_0 se puede obtener de la expresión anteriormente obtenida y es $Ve_0 = (1-0)(l-1) = l-1$.

La próxima propiedad garantiza que variando *p* se puede conseguir una distribución de E_l con cualquier varianza que se desee entre la mínima 0 y la máxima $l-1$.

Propiedad 3 Para todo elemento d de D_l existe un elemento e de E_l tal que la varianza de e es igual a la de d. En particular para la distribución binomial $b \in D_l$ de parámetros $n = l$ y $p = l^{-1}$, tenemos que $e_p \in E_l$ con $p = 1 - l^{-1}$ iguala a d en varianza.

Demostración 3 Como vimos, la varianza de los operadores de D_l alcanza mínimo 0 y máximo $l-1$ en operadores de E_l. Como la varianza de los operadores de E_l es $(1-p)(l-1)$, una función continua de p, debe tomar todos los valores intermedios, en particular el de la binomial $b \in D_l$ de parámetros $n = l$ y $p = l^{-1}$. Igualando varianzas podemos encontrar el valor de p para el que $V e_p = V b$.

Tenemos que

$$\begin{aligned}Vb &= ll^{-1}\left(1 - l^{-1}\right) \\ &= \frac{l-1}{l}.\end{aligned}$$

Luego p es tal que

$$\begin{aligned}(1-p)(l-1) &= \frac{l-1}{l} \\ 1 - p &= l^{-1} \\ p &= 1 - l^{-1}.\end{aligned}$$

La última propiedad ilustra un comportamiento que suele ser de interés, la capacidad de estos operadores al dispersar cromosomas repetidos.

Propiedad 4 El número aleatorio de cromosomas que cambian en uno o más genes al aplicar un operador con distribución $e_p \in E_l$ a una población de r cromosomas repetidos es binomial con parámetros $r\ y\ p\left(1 - l^{-1}\right) + l^{-1}$. Si el operador es el clásico binomial con probabilidad de mutación l^{-1} la cantidad de cromosmas que cambian es binomial con parámetros $r\ y\ \left(1 - l^{-1}\right)^l$.

Demostración 4 Se trata de r ensayos independientes con probabilidad de éxito en cada cromosoma igual a la probabilidad de que haya al menos una mutación. En el caso de $e_p \in E_l$ esta probabilidad es

$$\begin{aligned} p_1 + p_l &= p + (1-p)\, l^{-1} \\ &= p + l^{-1} - pl^{-1} \\ &= p\left(1 - l^{-1}\right) + l^{-1}. \end{aligned}$$

En el caso de número de mutaciones binomial con parámetro l^{-1} la probabilidad de que ningún gen de un cromosma mute es la de que en l ensayos independientes ocurre un evento de probabilidad $1 - l^{-1}$, es decir $\left(1 - l^{-1}\right)^l$.

4. Ensayos Computacionales

Realizamos ensayos computacionales para obtener la velocidad de convergencia al óptimo mediante algoritmos genéticos con el operador de mutación clásico y comparar esa velocidad con la del algoritmo que utiliza una de las distribuciones de la familia E_l. Se trata de ensayos de maten ática experimental caracterizados por las siguientes condiciones que los definen.

(1) El óptimo global es conocido, por lo cual podemos contar el número de generaciones que le demanda al algoritmo genético alcanzarlo. En nuestro caso adoptamos como problema de optimización el problema de la mochila (knapsack problem) el cual es resoluble exactamente por programación dinámica en tiempo pseudopolinomial. Esto representa tiempos cortos para las instancias relativamente pequeñas que utilizamos.

(2) Las instancias del problema utilizadas son generadas independientemente y al azar. En particular, para generar una instancia del problema de la mochila de longitud l utilizamos los pesos (1, 2, ..., l) y les asignamos una permutación

al azar (con todas las permutaciones igualmente probables) de los valores (10, 20, ..., 10l), donde el factor 10 sólo se introduce para no tener la misma escala que la de los pesos y poder monitorear visualmente los resultados con más facilidad. Para el tope del peso que admite la mochila consideramos el valor $\frac{1}{d}\sum_{i=1}^{l} i = \frac{1}{d} l(l+1)/2$ con $d = 2$, es decir, la suma total de pesos dividida por 2. Tomamos este valor si es entero y si no lo es tomamos el entero mayor más cercano.

(3) El algoritmo que utilizamos busca acentuar la relevancia del aspecto de interés, esto es, la mutación. Para ello prescindimos del operador de cruce. La selección se reduce a ordenar la población por fitness y copiar la mejor mitad sobre la otra mitad. Implementamos elitismo con tamaño de ´elite $e = 2$, es decir que en cada generación los dos cromosmas con más fitness no son mutados.

Utilizamos un tamaño poblacional igual a 50 con cromosomas de longitud $l = 20$. La población inicial es generada al azar con independencia e igual probabilidad de 0 y 1 para cada gen. Cada uno de nuestros ensayos compara el operador clásico binomial con probabilidad de mutación $\mu = l^{-1}$ con el operador que utiliza un elemento de la familia E_l para la cantidad de mutaciones con un cierto valor del parámetro p. Para cada

ensayo se realizan 300 pruebas. En cada prueba se genera un nuevo problema de la mochila, se lo resuelve mediante programación dinámica y se utiliza el óptimo para calcular el número de generaciones que demanda su obtención al algoritmo con distribución binomial y al que utiliza la distribución de E_l. En cada prueba se resta el número de generaciones utilizadas por el segundo operador a las del primero, de manera que cada valor positivo indica una velocidad mayor del segundo operador. El conjunto de diferencias se analiza estadísticamente (ver apéndice) con el estadístico $\frac{S_n - n\frac{1}{2}}{\sqrt{n\frac{1}{2}(1-\frac{1}{2})}}$ donde S_n es el número de diferencias positivas, $n = 300$ es el número de pruebas y que tiene distribución asintótica normal de parámetros 0 y 1. El resultado de los experimentos se resume en la siguiente tabla.

p	Diferencias positivas	Estadístico	p valor
1	150	0	0.5
0.95	157	0.8083	0.2095
0.9	152	0.2309	0.4087
0.85	165	1.7321	0.0416
0.8	170	2.3094	0.0105
0.75	173	2.6558	0.004
0.7	154	0.4619	0.3221
0.65	166	1.8475	0.0323
0.6	159	1.0392	0.1493
0.55	132	-2.0785	0.9812

Cuadro 1. Tabla de resultados para distintos valores de p.

5. Consideraciones sobre los Resultados de los Ensayos

Los datos de la tabla motivan el siguiente análisis.

- Con $p = 1$ tenemos varianza nula y los resultados son en promedio iguales con la distribución $e_p \in E_l$ comparados con el operador binomial. No haber obtenido resultados mejores puede explicarse porque la varianza nula restringe la exploración, en efecto, en cada generación cada cromosoma no llega más allá de una distancia de Hamming igual a 1 respecto de su posición anterior.
- Con $p = 1 - l^{-1}$ que es en nuestro caso $1 - 1/20 = 0{,}95$ tenemos, como vimos en la Propiedad 3, que se igualan las varianzas de la binomial y de nuestra distribución concentrada en tres puntos. Aún con varianzas iguales, las velocidades para la distribución $e_p \in E_l$ mejoran a las de la binomial sugiriendo la conveniencia de que algunos cromosomas se dirijan cada tanto y con una probabilidad no demasiado pequeña, a una zona lejana del espacio de búsqueda, respecto de la zona actualmente explotada. Posteriores mutaciones en un gen se encargan de la difusión hacia posiciones vecinas cercanas consiguiéndose un balance satisfactorio entre exploración y explotación.
- Todos los ensayos para p entre 0,6 y 0,95 dan resultados mejores para e_p que

para la binomial. En varios casos esta diferencia es estadísticamente significativa, como lo indican los $p-$ valores.

- Para valores de p iguales a 0,5 o menores la cantidad de generaciones es muy grande para el algoritmo gen ético con cantidad de mutaciones dada por la distribución e_p y no llega a detenerse en tiempo razonable. Aparentemente la exploración se hace casi imposible por la alta probabilidad de que un cromosoma sea mutado en todos sus genes. En el caso extremo en el que $p = 0$ tenemos que la probabilidad de tener cero mutaciones es $1 - 1/l$, la de mutar todos los l genes es $1/l$ y las demás probabilidades son cero. Esto significa, dado que estos algoritmos no tienen cruce, que si el óptimo global no se obtuvo en la población inicial o no es el resultado de mutar todos los genes de algún cromosma de la población inicial, es imposible que se obtenga.
- En este trabajo no abordamos la resolución de problemas reales ni problemas de prueba con una dificultad particular. Sin embargo, dado que en cada una de las 300 pruebas de cada uno de los ensayos correspondientes a los distintos valores de p se genera un problema al azar, los resultados sugieren que en un problema real puede ser conveniente utilizar un operador de mutación que mantiene el valor esperado de mutaciones (determinado por la probabilidad de mutación que se considera conveniente para ese problema) pero aumenta la varianza respecto de la distribución binomial.

6. Apéndice

6.1 Estadístico para el test de hipótesis y el cálculo de p-valores

Con el fin de llevar a cabo un test estadístico de la hipótesis de que un operador alternativo (método 1) mejora la velocidad del algoritmo gen ético con respecto al clásico binomial (método 0), planteamos las siguientes hipótesis nula y alternativa:

H_0: La velocidad de ambos algoritmos es igual versus H_1: El segundo operador es mejor.

Si p es la probabilidad de que la velocidad en un ensayo sea mejor con el método 1 que con el 0 las hipótesis son:

$H_0: p = 1/2$ versus $H_1: p > 1/2$

Si hicimos n pruebas podemos definir para $i \in \{1, 2, ..., n\}$ las variables

G^0_i, G^1_i: Número de generaciones para llegar al ´optimo por parte de los métodos 0 y 1 en la prueba i.

$D_i = G^0_i - G^1_i$. Es decir que $D_i > 0$ significa que el método clásico demoró más que la mezcla.

Además definimos para cada prueba i las variables indicadoras $X_i = 1$ tales que $X_i = 1$ si $D_i > 0$ y $X_i = 0$ en caso contrario, y por lo tanto $X_i \sim Bi(1,p)$ y las X_i son independientes.

Sea $S_n = \sum_{i=1}^{n} X_i$. Entonces tenemos $p = E\,X_i = E\,\overline{X_n}$, $S_n \sim Bi(n, p)$, $ES_n = np$ y $V\,S_n = np(1 - p)$.

Luego bajo H_0 se cumple que el estadístico $\frac{S_n - n\frac{1}{2}}{\sqrt{n\frac{1}{2}(1-\frac{1}{2})}}$ tiene distribución $N(0,1)$ aproximadamente por Teorema Central del Límite.

Referencias

1. De Falco, I., Della Cioppa, A., Tarantino, E.: Mutation-Based Genetic Algorithm: Performance Evaluation. Applied Soft Computing. 1. 285–299 (2002)
2. De Jong, K. An Analysis of the Behavior of a Class of Genetic Adaptive Systems.Ph.D. Dissertation, Department of Computer and Communication Sciences, University of Michigan, Ann Arbor (1975)
3. Di Rado, A., Podestá, G. Propuesta de implementación y análisis de convergencia para algoritmos genéticos binarios con control de diversidad. Tesis de licenciatura en Ciencias de la Computación, Universidad de Buenos Aires (2014)
4. Sidaner, A., Bailleux, O., Chabrier, J.J. Measuring the Spatial Dispersion of Evolutionist Search Processes: Application to Walksat. In: 5th International Conference, Evolution Artificielle, EA 2001 At: Le Creusot FranceVolume: LNCS 2310 Springer 2002 pp 77–87 (2001)

Coordinadores:

- Guillermo Leguizamón (UNSL)
- Carolina Salto (UNLPam)
- Waldo Hasperué (UNLP)

CACiC 2023

CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

XXIV Workshop de Agentes y Sistemas Inteligentes

> Short Papers

WASI

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Utilizando Argumentación Rebatible en la Detección y Respuesta ante Intrusión en Sistemas Biométricos

Graciela R. Etchart[1], Juan C.L. Teze[1], Carlos E. Alvez[1],
María V. Martinez[2], Gerardo I. Simari[3]

[1]Facultad de Ciencias de la Administración, Universidad Nacional de Entre Ríos (UNER),
[2]Instituto de Investigación en Ciencias de la Computación, Universidad Nacional de Buenos Aires (UBA), Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET),
[3]Instituto de Ciencias e Ingeniería de la Computación (UNS-CONICET), Departamento de Ciencias e Ingeniería de la Computación, Universidad Nacional del Sur (UNS)
[1]{graciela.etchart, carlos.alvez, carlos.teze}@uner.edu.ar, [2]mvmartinez@dc.uba.ar,
[3]gis@cs.uns.edu.ar

Abstract. En las últimas décadas, la comunidad biométrica ha investigado la seguridad de los sistemas de reconocimiento biométrico identificando puntos de ataques potenciales, estudiando posibles amenazas y proponiendo contramedidas para superarlas. En este sentido, diversos trabajos apuntan a resolver situaciones de ataques dirigidos a la plantilla (o *template*) biométrica. En menor medida, se han encontrado propuestas orientadas a la detección y mitigación de intrusiones en el canal de comunicación del sistema biométrico. Para aumentar la seguridad en este punto, las técnicas de detección de intrusión son considerablemente útiles. En este trabajo se busca realizar la detección y respuesta ante ataques utilizando argumentación rebatible, con el propósito de proporcionar una estructura que favorezca una toma de decisiones de seguridad más informada. Se considera que el enfoque formal que brinda la argumentación complementará sustancialmente los sistemas de seguridad existentes en sistemas biométricos.

Keywords: Argumentación, Detección de intrusión, Sistemas biométricos.

1. Introducción

Los sistemas biométricos tienen varias ventajas con respecto a los métodos de autenticación tradicionales. Sin embargo, son vulnerables a ataques que pueden comprometer su seguridad y privacidad [1]. Varios trabajos han identificado y caracterizado puntos de ataques potenciales en los sistemas biométricos [2, 3]. Los puntos vulnerables se pueden agrupar en: ataques dirigidos hacia los equipos físicos y ataques orientados al canal de comunicación. En la literatura se han propuesto diferentes contramedidas para evitar o minimizar los riesgos derivados de los ataques de adversarios. Diversos trabajos apuntan a resolver situaciones de ataques dirigidos a la plantilla (o *template*) biométrica. En menor medida, se han encontrado propuestas dirigidas a la detección y mitigación de intrusión en el canal de comunicación del sistema biométrico.

Dentro del campo de la seguridad en redes, los sistemas detectores de intrusión (IDS, por sus siglas en inglés) son una herramienta de creciente preponderancia [4]. Estos sistemas procuran hallar acciones que puedan comprometer la confidencialidad, la integridad o la disponibilidad de un recurso. Los IDSs desde sus inicios han sido

objeto de una constante revisión y evolución. En relación a los procesos biométricos, la detección de intrusión puede ser utilizada como una capa de defensa contra los ataques que surgen en el canal de comunicación. Si se detecta la intrusión, se puede iniciar una advertencia y recomendar medidas para prevenir o minimizar los daños en el sistema.

Aunque numerosos IDSs han sido estudiados y efectivamente aplicados en diferentes escenarios de la vida real [5-6], la naturaleza dinámica del dominio en donde se producen los ataques en ocasiones conduce a situaciones de información en conflicto, principalmente cuando se razona en presencia de información incompleta y potencialmente contradictoria. En este escenario, los sistemas argumentativos [7, 8] pueden verse como una alternativa útil para hacer frente a estas cuestiones. El mecanismo de inferencia sobre el cual están basados permite decidir entre conclusiones contradictorias y adaptarse fácilmente a entornos cambiantes.

En inteligencia artificial, el área de argumentación computacional se especializa en modelar el proceso de razonamiento qué típicamente realizan las personas cuando buscan establecer qué conclusiones son aceptables en un contexto de desacuerdo. En el marco de este trabajo, el proceso de razonamiento permite filtrar selectivamente información para detectar amenazas, sugerir acciones y acelerar la respuesta ante la presencia de una posible intrusión.

2. Aportes realizados

Este trabajo se centra en introducir una arquitectura que extiende las capacidades de razonamiento de los sistemas biométricos incorporando argumentación al proceso de y respuesta a intrusiones. En la solución propuesta se utiliza un servicio de razonamiento basado en argumentación, para representar el conocimiento e inferir la acción de respuesta adecuada frente a una intrusión detectada en el canal de comunicación de un sistema biométrico, procurando minimizar el impacto del ataque en el funcionamiento habitual del sistema.

A continuación, se presenta el diseño arquitectónico del Sistema de Respuesta a Intrusión Basado en Argumentación (SRIBA) propuesto y una descripción de las operaciones del mismo, recurriendo a los formalismos Unified Modeling Language (UML) y modelo C4 de visualización de arquitectura de software[1]. El SRIBA presenta una arquitectura modular donde se pueden identificar tres contenedores bien diferenciados: el Escáner, el Gestor de Consulta Contextual (GCC) y el Razonador. En la Figura 1 se brinda una vista a alto nivel del sistema y se especifica el sistema externo con el cual interactúa el SRIBA (en este caso, el sistema biométrico). También se identifica la interacción con el usuario final que hará uso de la funcionalidad del sistema.

El Escáner es el responsable de obtener y registrar información del canal de comunicación monitorizado, analizando paquetes capturados en segmentos de red que conforman el sistema biométrico y detectando intrusiones al mismo. Uno de los componentes del Escáner es un sistema de detección de intrusión de uso indebido

[1] https://c4model.com/

basado en red, capaz de detectar accesos no autorizados utilizando un conjunto de reglas predeterminadas, y emitir alertas de posibles intrusiones.

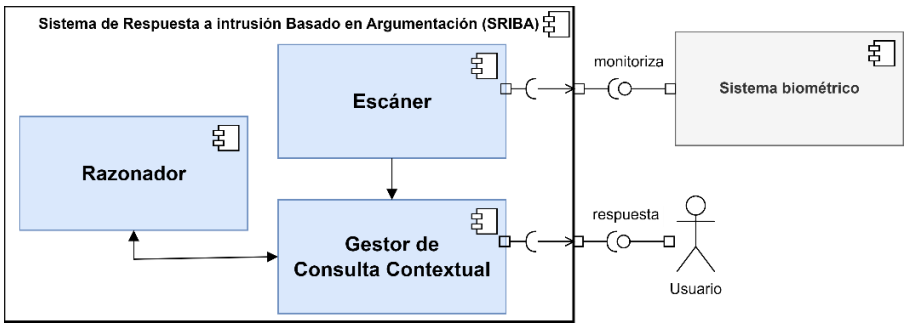


Fig. 1. Arquitectura del SRIBA

Por otro lado, el GCC tiene como responsabilidad procesar la información obtenida por el Escáner y generar hechos que expresan observaciones de los datos contenidos en las alertas. Estos hechos conforman el contexto enviado al componente razonador junto a la consulta. Una vez recibida la respuesta, el GCC la despliega al usuario del sistema.

En último lugar, el Razonador es la parte central de la arquitectura. Esta componente brinda un servicio de razonamiento argumentativo. Una propuesta para su implementación se enfoca en un sistema argumentativo en particular, denominado Programación en Lógica Rebatible (DeLP, por sus siglas en inglés) [9]. DeLP combina resultados de programación en lógica y argumentación rebatible, y posee flexibilidad en cuanto a su aplicación en diferentes dominios, con el propósito de proporcionar una estructura que favorezca una toma de decisiones de seguridad más informada. La argumentación rebatible proporciona un enfoque formal para manejar la inconsistencia de los datos que deben utilizarse en la toma de decisiones y permite extraer un conjunto coherente de reglas que puedan aplicarse para llegar a una decisión. Además, los argumentos que se construyen permiten explicar los resultados del razonamiento a los responsables de la toma de decisiones humanas, de manera que se aclare la situación y se mejore la calidad de las decisiones.

Como prueba de concepto para favorecer la comprensión del funcionamiento del Razonador, se realizan dos instanciaciones utilizando formalismos concretos para representar el conocimiento, las consultas y las respuestas. Para una instanciación se utiliza el concepto formal de servidor de razonamiento en Programación en Lógica Rebatible o DeLP-Servers [10, 11]. Los DeLP-Servers presentan un enfoque cliente-servidor caracterizado por permitir representar información o conocimiento público y responder consultas utilizando dicho conocimiento público junto con información privada provista por el cliente. De esta manera, el Razonador procesa la consulta recibida utilizando el contexto brindado por el GCC y, mediante un proceso argumentativo, brinda como respuesta la contramedida para contrarrestar los efectos de una intrusión detectada. Por otra parte, con el propósito de posibilitar la toma de decisiones multicriterio para la selección de la contramedida a desplegar frente a una intrusión, se instancia el proceso de razonamiento siguiendo un formalismo basado en reglas condicionales introducido por Burón Brarda et al. [12], el cual permite codificar

las preferencias entre criterios de comparación de acciones de respuestas en términos de requisitos y valores de tolerancia. En otras palabras, esta instanciación permite llevar a cabo el proceso de elección de la contramedida adecuada a partir de un proceso de decisión multicriterio basado en argumentación. De esta manera, es posible dar prioridad a aquellos atributos que mejor definen una recomendación relevante, en función de las políticas de seguridad que se deseen aplicar y diversas alternativas de acciones de respuesta. Una característica común en ambos formalismos y de gran relevancia, es que las conclusiones obtenidas pueden explicarse fácilmente mediante el proceso de razonamiento argumentativo, el cual permite ofrecer explicaciones como conjuntos coherentes de razones a favor o en contra de la acción de respuesta recomendada. De esta manera, se favorece una toma de decisión de seguridad más informada.

Finalmente, como se mencionó con anterioridad, en el SRIBA el Escáner opera mediante un sistema detector de intrusión basado en uso indebido. En este tipo de sistemas el costo de desarrollo y mantenimiento de los conjuntos de reglas constituye un importante problema. Con el propósito de mejorar la actualización de la base de datos de reglas para la detección (firmas) de esta componente, se realizó un estudio preliminar sobre distintas herramientas de aprendizaje automático. La motivación fue dar unos primeros pasos para proponer una solución aceptable para generar firmas orientadas a la detección personalizadas para el sistema biométrico. En esta tarea se estudió SIRUS (Stable and Interpretable RUle Set[2]), un nuevo algoritmo de reglas para la clasificación. Este algoritmo brinda un nivel útil de interpretabilidad en los modelos que genera, necesario para su traducción a una firma entendible por el sistema detector de intrusión.

3. Posibles líneas de investigación a futuro

En el contexto de esta investigación los resultados son preliminares pero promisorios en cuanto a su aplicación efectiva en el contexto de aplicación en el mundo real. La arquitectura presentada constituye una propuesta a alto nivel, perfectible en trabajos posteriores. Además, se considera que esta propuesta supone un avance para conseguir un proceso de reconocimiento más seguro y una toma de decisiones sobre seguridad más informada; sin embargo, la investigación debe seguir persiguiendo garantizar la confidencialidad, la integridad y la disponibilidad de un sistema biométrico.

El SRIBA parte de una arquitectura general que puede ser modularizada e implementada de diversas maneras. Además, este enfoque puede aplicarse también a otros sistemas tecnológicos, y su utilidad no se limita a los sistemas biométricos.

En la búsqueda de soluciones y mejoras a las diferentes cuestiones planteadas, han surgido nuevas ideas e inquietudes que constituyen algunas líneas de investigación. Uno de estos trabajos futuros consiste en estudiar cómo aprovechar los conceptos propuestos para desarrollar aplicaciones con otros formalismos de razonamiento, como podría ser ontologías. Otra tarea consiste en lograr un comportamiento proactivo en el

[2] https://cran.r-project.org/web/packages/sirus/

SRIBA. Para esto es necesario utilizar algoritmos de predicción de ataques de tal manera que el sistema de respuesta pueda ejecutar la acción de respuesta frente a una intrusión antes que ésta se lleve a cabo. Una vez que el tipo de ataque ha sido predicho, la inferencia de la respuesta adecuada se realizaría siguiendo el proceso de razonamiento propuesto en este trabajo. También, es posible profundizar el estudio de métricas de seguridad y de metodologías utilizadas para su obtención, a fin de ampliar la arquitectura propuesta con módulos que gestionen la obtención de dichas métricas.

Referencias

1. Rui, Z., and Yan, Z.: A Survey on Biometric Authentication: Toward Secure and Privacy-Preserving Identification. IEEE Access, 7, 5994-6009 (2019).
2. Ratha, N., Connell, J., and Bolle, R.: Enhancing security and privacy in biometrics-based authentication systems. IBM Systems Journal, 40, 614-634 (2001).
3. Jain, R., and Kant, C.: Attacks on Biometric Systems: An Overview. International Journal of Advances in Scientific Research, 1, 283-288 (2015).
4. Jaiganesh, V., Mangayarkarasi, S., and Sumathi, D.P.: Intrusion detection systems: A survey and analysis of classification techniques. International Journal of Advanced Research in Computer and Communication Engineering, 2 (2013).
5. Hamed, T., Ernst, J.B., and Kremer, S.C.: A Survey and Taxonomy of Classifiers of Intrusion Detection Systems. In Daimi K. (eds) Computer and Network Security Essentials. Springer, Cham, 21-39 (2018).
6. Ozkan-Okay, M., Samet, R., Aslan, Ö., and Gupta, D.: A Comprehensive Systematic Literature Review on Intrusion Detection Systems. In IEEE Access, 9, 157727-157760 (2021).
7. Paredes J., Simari G.I., Martinez M.V., Falappa, M.: Detecting malicious behavior in social platforms via hybrid knowledge- and data-driven systems, Future Generation Computer Systems, 125, 232-246 (2021)
8. Leiva, M., García, A., Shakarian, P. and Simari, G.I.: Argumentation-Based Query Answering under Uncertainty with Application to Cybersecurity. Big Data Cogn. Comput., 6, 91 (2022).
9. García, A. and Simari, G.R.: Defeasible logic programming: An argumentative approach. Theory and Practice of Logic Programming (TPLP), 4, 95-138 (2004).
10. García, A., Rotstein, N.D., Tucat, M., and Simari, G.R.: An Argumentative Reasoning Service for Deliberative Agents. In Zhang, Z., Siekmann, J. (eds) Knowledge Science, Engineering and Management. KSEM 2007. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 4798, 128-139 (2007).
11. Tucat, M.: Grupos de Servicios de Razonamiento para el Procesamiento de Consultas Contextuales en Paralelo. PhD thesis, Universidad Nacional del Sur, Bahía Blanca, Argentina. (2011).
12. Burón Brarda, M.E., Tamargo, L.H., and García, A.: An approach to enhance argument-based multi-criteria decision systems with conditional preferences and explainable answers. Expert Systems with Applications, 126, 171-186 (2019).

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Adapting PALS to solve VRP with Time Windows

C. Bermudez[1][0000-0002-1226-3735], H. Alfonso[1][0000-0001-6023-5299], G. Minetti[1][0000-0003-1076-6766], and C. Salto[1,2][0000-0002-3417-8603]

[1] Facultad de Ingeniería, Universidad Nacional de La Pampa, Argentina
[2] CONICET, Argentina
bermudezc,alfonsoh,minettig,saltoc@ing.unlpam.edu.ar

Abstract. In this article, we present a search technique based on metaheuristics for approximately solving the vehicle routing problem with time windows (VRPTW). The Problem Aware Local Search (PALS) is the metaheuristic selected, which proves to be a powerful technique for solving the DNA Fragment Assembly Problem. PALS performs a trajectory in the search space by improving a single solution using a specific move operator. A detailed description of the algorithmic design points and the operators for the VRPTW are carried out. Furthermore, preliminary results are shown and analyzed.

Keywords: Vehicle Routing Problem with Time Windows, Optimization, Problem Aware Local Search (PALS)

1. Introduction

Vehicle Routing Problems (VRP) [1] have been a subject of active research in the optimization community for the last 60 years. Since then, many changes in the initial problem formulation have been made. In this sense, a well-known variant of the VRP incorporates Time Windows (VRPTW), which is present in many real-world scenarios, such as in distribution operations, where the time window of the delivery is a crucial parameter of the problem. The VRPTW consists in delivering goods to a set of customers with known demands through minimumcost vehicle routes, beginning and finishing at the depot. Each customer is to be serviced once by only one vehicle. In addition, each vehicle has a limited capacity, and each customer has an associated time window. This last restriction indicates that each customer provides a time frame within which a particular service or task must be completed (for example, loading or unloading a vehicle). A vehicle may arrive early, but it must wait until the start of service time.

Regarding the practical relevance of VRP and its NP-hardness [1], solving this problem and its variants are of great interest for both the research community and logistics and transportation companies. Therefore, heuristic algorithms have been proposed, mainly divided into construction and local search heuristics [2,3]. In contrast to heuristics, metaheuristics intend to escape from local optimum, as they explore a more significant subset of the solution space [4,5,6]. In particular, in this work, a trajectory-based metaheuristic known as Problem Aware Local Search (PALS) [7] is used to solve the VRPTW. PALS performs a trajectory in the search space by improving a single-solution using a specific local search. The PALS main strength is to estimate variations on the fitness values, without having to calculate the complete fitness function for each point in the neighborhood. Therefore, a very inexpensive and systematic explorative search over the whole neighborhood is efficiently carried out in each iteration. Then only the candidate neighbors are stored, exploiting the promising

search areas. After this, PALS selects one of the candidates and this is applied to the solution. This process continues until no more candidate neighbors are generated.

PALS was designed for solving the DNA Fragment Assembly Problem. In [8], the authors extended its application to other NP-hard optimization problems. The good results found in [8] motivated the adaptation of PALS to the VRPTW problem. The work's contribution is centered around the adaptation of PALS to the features exhibited for the VRPTW, as a permutation-based combinatorial optimization problem. Consequently, the solution representation, the move operator, the criteria to select a move, and more details are described in following sections. Finally, a discussion of the research state is carried out.

2. Problem Formulation

The time dependent vehicle routing problem with hard time windows studied in this research can be described as follows. Let a set of identical vehicles denoted by K and by a direct network $G(N,A)$, where N is the set of nodes and $A = (i,j) : i \neq j, i, j \in N$ is the set of arcs. Node 0 represents the central depot, while $N^* = \{N/0\}$ denotes the customers. Every arc (i,j), which is a path from node i to node j, is characterized by the distance d_{ij}. The value t_{ij} stands for the travel time from node i to node j and has the same value with d_{ij}, as the assumption that each distance unit corresponds to one specific time unit is made. Each customer i is represented by the demand q_i, the service time needed s_i and the time window (e_i, l_i), where e_i is the opening time and l_i the closing time. Customers must be served by exactly one vehicle that may arrive any time within the time window. The arrival time is given by t_i. Additionally, if the vehicle arrives before the beginning of the time window (e_i), it must wait for w_k^i time, until service is possible, while no vehicle may arrive after the end of the time interval (l_i).

The VRPTW goal is to service all customers, minimizing the number of vehicles and the total time or traveled distance while simultaneously ensuring that all the constraints are satisfied. The mathematical model of the VRPTW, presented in equations 1 to 9, uses $x_{i,j}^k$ as decision variable that is equal to 1 if vehicle k drives from node i to node j, and 0 otherwise.

First Objective: $\quad minimize \sum_{k \in K} \sum_{j \in N^*} x_{0j}^k \qquad (1)$

Second Objective: $\quad minimize \sum_{k \in K} \sum_{(i,j) \in A} d_{ij} x_{ij}^k \qquad (2)$

subject to:

$$\sum_{i \in N} q_i \sum_{j \in N} x_{ij}^k \leq q_{max}^k, \forall k \in K (3) \qquad \sum_{k \in N} \sum_{j \in N} x_{ij}^k = 1, \forall i \in N (4)$$

$$\sum_{j \in N^*} x_{0j}^k = 1, \forall k \in K (5) \qquad \sum_{i \in N} x_{il}^k - \sum_{j \in N} x_{lj}^k = 0, \forall l \in N, \forall k \in K (6)$$

$$\sum_{j \in N^*} x_{j0}^k = 1, \forall k \in K (7) \qquad t_0 = w_0^k = s_0 = 0 (8)$$

$$\sum_{j \in N} x_{ij}^k (t_i + w_i^k + s_i + t_{ij}) \leq t_j, \forall i \in N, \forall k \in K (9)$$

3. Problem Aware Local Search (PALS)

The pseudo-code of PALS is shown in Algorithm 1. A random initial permutation, *s*, is created in order to represent a solution. In each iteration, the current permutation is the base to generate systematically all possible movements by applying the move operator. For each movement, PALS computes Δ_f as the difference between the partial fitness evaluations, which are calculated before and after the move operator application by means of the **`CalculateDelta`** function. The set *L* is formed by the movements with $\Delta_f \leq 0$ (called candidate movements). The best candidate is selected from *L* and is applied to the current solution, starting the cycle again until no possible changes. The PALS adaptation to the VRPTW problem results in the customization of design points that are specific aspects of the problem, such as solution representation, move operators, fitness function, and **`CalculateDelta`** function. We also redefine two algorithmic design points as seeding strategy and criterion to select a movement from *L*.

Representation. The solution representation must be suitable and relevant to the tackled optimization problem. We apply one widely used in the literature that consists of a list of routes, where each one is a permutation of integers that indicates a sequence of customers represented by a unique integer ID. Consequently, a VRPTW solution encoded by a list of permutations must satisfy the following conditions to be legal: all customers must be present in the list, and no duplicate customers are allowed (Eqs. 4 to 7).

Algorithm 1 Problem Aware Local Search

```
initialize s; {generate the initial solution}
repeat
L = ∅;
for i = 0 to N do
for j = 0 to N do
Δf = CalculateDelta(s,i,j);
if Δf ≤ 0 then
L = L∪ <i,j,Δf>; {Add candidate movements to L}
    end if
  end for
end for
if L <> ∅ then
<i,j,Δf> = Extract(L); {Select a movement from L}
ApplyMovement(s,i,j); {Modify the solution}
end if
until no changes return s;
```

Move Operator. The move operator is an adaptation of the Interchange operator for this solution representation, where the client *i* assigned to route *k* can interchange its visit turn with a client *j* belonging to the same route or another one.

Fitness function. Our definition takes into account the second VRPTW objective function (Eq. 2) and customer time windows, (e_i,l_i), applying an adaptive penalization [9] wherever the vehicle arrives after the closing time and repairing the solution when the maximum vehicle capacity (q^k_{max}) is exceeded.

CalculateDelta function. This function is directly related to the move operator and the fitness function. As a result, we define CalculateDelta function by applying partially the fitness function only on the modified routes.

Seeding Strategies. We propose an heuristic seeding strategy that considers the q^k_{max} and the whole demand ($\sum_{i \in N} q_i$) to determine the minimum number of vehicles (or routes), attending the first VRPTW objective and calculated as the Eq.10 presents. After that, the customers are randomly selected and inserted in a route, which is chosen in uniform way.

$$\frac{q^k_{max}}{\sum_{i \in N} q_i} + 1 \tag{10}$$

Criteria to select movements from *L*. Our PALS version applies uniformly the original criterion that extracts the *best* candidate movement or the criterion that chooses a candidate in a *random* way [8]. In this way, we avoid the search stagnating in local optima.

4. Discussion

In this research phase, we conduct tests on the proposed PALS using a benchmark widely used in the literature, as Solomon's instances [3]. Fig. 1 shows the PALS behavior considering the variations of the fitness and objective functions during the search. As both functions exhibit different value ranges, we plot their evolution on the x-axis but with distinct scales on the left and right y-axes; therefore, the fitness values are shown on the left, while the objective ones are displayed on the right. In this sense, we observe a significant reduction in fitness function for small, middle, and large instances (with 25, 50, and 100 customers, respectively), but a much smaller decrease in the objective function. Our future research step is to improve this algorithm to find the optimal solutions and avoid search stagnation.

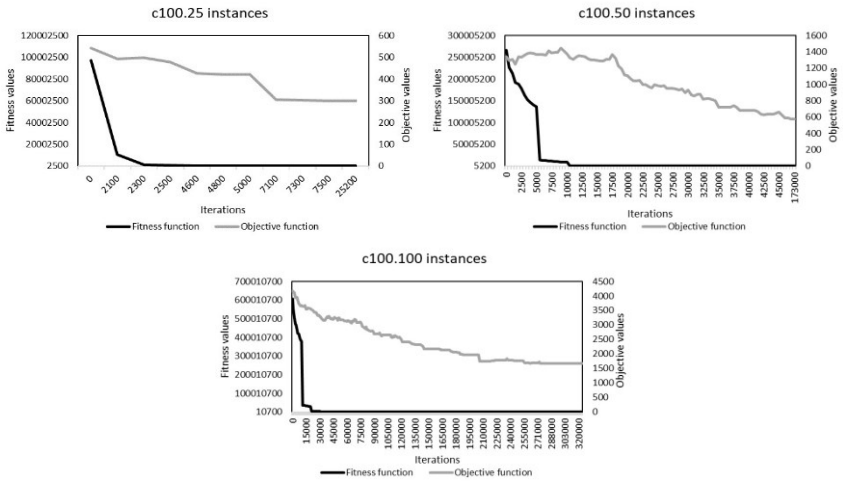


Fig.1: Fitness and objective functions evaluated to C100 instances with 25, 50, and 100 customers, respectively.

References

1. J. K. Lenstra and A. H. G. R. Kan, "Complexity of vehicle routing and scheduling problems," *Networks*, vol. 11, pp. 221–227, 1981.
2. O. Bräysy and M. Gendreau, "Vehicle routing problem with time windows, part i: Route construction and local search algorithms," *Transportation Science*, vol. 39, no. 1, pp. 104–118, 2005.
3. M. M. Solomon, "Algorithms for the vehicle routing and scheduling problems with time window constraints," *Operations Research*, vol. 35, no. 2, pp. 254–265, 1987.
4. T.-C. Chiang and W.-H. Hsu, "A knowledge-based evolutionary algorithm for the multiobjective vehicle routing problem with time windows," *Computers Operations Research*, vol. 45, pp. 25–37, 2014.
5. G. D. Konstantakopoulos, S. P. Gayialis, E. P. Kechagias, G. A. Papadopoulos, and I. P. Tatsiopoulos, "A multiobjective large neighborhood search metaheuristic for the vehicle routing problem with time windows," *Algorithms*, vol. 13, no. 10, 2020.
6. A. Dixit, A. Mishra, and A. Shukla, "Vehicle routing problem with time windows using meta-heuristic algorithms: A survey," in *Harmony Search and Nature Inspired Optimization Algorithms*, N. Yadav, A. Yadav, J. C. Bansal, K. Deep, and J. H. Kim, Eds. Singapore: Springer Singapore, 2019, pp. 539–546.
7. E. Alba and G. Luque, "A new local search algorithm for the DNA fragment assembly problem," in *Evolutionary Computation in Combinatorial Optimization, EvoCOP'07*, ser. Lecture Notes in Computer Science, vol. 4446. Valencia: Springer, 2007, pp. 1–12.
8. G. F. Minetti, G. Luque, and E. Alba, "The problem aware local search algorithm: an efficient technique for permutation-based problems," *Soft computing*, pp. 1–14, 2017.
9. M. Gendreau, A. Hertz, and G. Laporte, "A tabu search heuristic for the vehicle routing problem," *Mgmt Sci*, vol. 40, pp. 1276–1290, 10 1994.

Aprendizaje no supervisado combinando autoencoders y agrupamiento

Lucas Cochella[1] [0009-0007-8333-8080], Laura Lanzarini[1,2] [0000-0001-7027-7564], and Patricia Jimbo Santana[3] [0000-0001-7432-1622]

Facultad de Informática. Universidad Nacional de La Plata
lucas.cochella@econo.unlp.edu.ar
Instituto de Investigación en Informática LIDI (Centro CICPBA)
laural@lidi.info.unlp.edu.ar
Facultad de Ciencias Administrativas Universidad Central del Ecuador
prjimbo@uce.edu.ec

Resumen Las técnicas de agrupamiento son las más utilizadas a la hora de resolver tareas descriptivas. Su entrenamiento no supervisado requiere del uso de alguna medida de distancia para identificar similitudes entre los datos disponibles. La cantidad de características utilizadas suele tener un impacto directo en la calidad del agrupamiento por lo que contar con una representación adecuada de los datos es de sumo interés. Un área de investigación reciente en el aprendizaje no supervisado es la combinación del aprendizaje de representación con redes neuronales profundas y la agrupación de datos. Este artículo presenta una línea de investigación relacionada con el estudio, diseño e implementación de un modelo que combine autoencoders con distintas técnicas de agrupamiento para analizar datos estructurados con alta dimensionalidad. Si bien los autoencoders han demostrado ser efectivos para aprender representaciones adecuadas a la hora de procesar imágenes, aún quedan aspectos por resolver cuando se trata de trabajar con datos estructurados. El ámbito de aplicación son los datos académicos de los estudiantes de la Universidad Central del Ecuador entre 2018 y 2022. Estos datos pueden contener información valiosa sobre el desempeño y la evolución de los estudiantes, así como posibles casos de error, fraude o corrupción que afecten a la calidad educativa. El agrupamiento basado en la representación aprendida puede ayudar a identificar estos casos, así como a descubrir patrones y tendencias que permitan mejorar el sistema educativo.

Keywords: Autoencoders · Agrupamiento · Reducción de Dimensionalidad · Deep Learning.

1. Introducción

El aprendizaje no supervisado es una modalidad de aprendizaje automático que busca extraer conocimiento de los datos sin contar con etiquetas o clases previas. Una de sus aplicaciones más frecuentes es el agrupamiento de datos, que consiste en asignar los datos a grupos o clústeres según algún criterio de similitud o distancia.

Para realizar el agrupamiento se requiere una representación adecuada de los datos que capture las características relevantes para la tarea sin incrementar excesivamente la dimensión del espacio de entrada. Una forma de obtener esta representación es mediante el uso de redes neuronales profundas, que son modelos computacionales inspirados en

el funcionamiento del cerebro humano, capaces de aprender representaciones complejas y no lineales a partir de los datos.

Dentro de las redes neuronales profundas, los autoencoders son redes que se entrenan para reconstruir los datos de entrada a partir de una representación interna más compacta y significativa. Esta representación se puede utilizar luego para realizar el agrupamiento con distintas técnicas.

Este trabajo se enmarca en un proyecto de Tesis de culminación de la Maestría en Inteligencia de Datos orientada a Big Data de la Facultad de Informática en la Universidad Nacional de La Plata y tiene por objetivo analizar el potencial de la combinación de métodos de agrupamiento no supervisados con técnicas de deep learning como los diferentes tipos de autoencoders. Como resultado de esta investigación se espera aportar una solución al problema de agrupar datos estructurados con alta dimensionalidad mediante la reducción que ofrecen los modelos de autoencoders. A continuación se presenta una revisión bibliográfica sobre los temas a analizar y una breve descripción del caso de estudio.

2. Trabajos relacionados

El aprendizaje profundo es una técnica basada en redes neuronales artificiales capaz de aprender representaciones complejas y no lineales a partir de los datos. En este trabajo, se propone combinar el aprendizaje de representación con autoencoders y las técnicas de agrupamiento para analizar el progreso académico de los estudiantes universitarios a partir de sus registros personales.

En relación a la aplicación de algoritmos de agrupamiento, que son métodos no supervisados que tienen como objetivo dividir un conjunto de datos en grupos o clústeres según algún criterio de similitud o distancia, encontramos a Ahmad Fikri Mohamed Nafuri et al (2022) [1] que analiza los datos comparando la performance de K-Means, BIRCH y DBSCAN realizando una selección de los atributos preponderantes de los datos con el fin de evitar las complicaciones que trae la multidimensionalidad al análisis. Valarmathy & Krishnaveni [2] por su parte van un paso más allá y realizan agrupamiento de datos comparando el rendimiento de 8 técnicas como Expectation Maximization, CLOPE, DBSCAN, entre otras.

Por otra parte, los autoencoders son un tipo especial de redes neuronales profundas que pueden aprender una representación comprimida y reconstruir los datos originales a partir de ella. Dentro de los trabajos relacionados con esta propuesta podemos mencionar a Tschannen et al. (2018) [3] quienes realizan una revisión en profundidad de los avances recientes en el aprendizaje de representación con autoencoders, y analizan el aprendizaje de representación desde la perspectiva de la teoría de la tasa-distorsión. Huang et al. (2023) [4] propone un método de agrupamiento profundo basado en un autoencoder embebido, que incorpora un autoencoder en la unidad codificadora y decodificadora del autoencoder prototipo, experimentan con múltiples conjuntos de datos de imágenes y muestran que su método tiene un rendimiento superior al de otros métodos existentes. Lu & Li (2022) [5] presentan un marco generalizado de aprendizaje de representación basado en autoencoders para el

agrupamiento de datos, y analizan diferentes tipos de datos. Muestran que su enfoque mejora el rendimiento del algoritmo K-Means.

El trabajo de Kanazawa et al. (2019) [6] se basa en un estudio sobre el análisis de agrupamiento de características basado en la representación de la capa oculta de un autoencoder. Aplican su método a datos sintéticos y reales, y evalúan la calidad del agrupamiento mediante el índice Davies-Bouldin. En el trabajo de Jimbo et al (2023) [7] se realiza una revisión sistemática de la literatura, considerando las investigaciones que se han desarrollado utilizando técnicas de inteligencia artificial para analizar el rendimiento académico en instituciones de educación superior. Aquí se identifican las principales técnicas utilizadas, como minería de datos, aprendizaje automático, sistemas expertos o redes neuronales, así como las variables y los indicadores empleados para medir el rendimiento académico.

Finalmente, Florent Forest, et al (2021) [8], combina la aplicación de la técnica de aprendizaje no supervisado Self Organizing Map con la aplicación de autoencoders para reducir la dimensionalidad de los datos obteniendo resultados alentadores mediante la implementación de DESOM en datasets compuestos por imágenes y texto.

3. Caso de estudio

Como caso de estudio, se propone analizar los datos de los estudiantes de la Universidad Central del Ecuador (UCE) entre 2018 y 2022.

La UCE es la universidad más antigua y la segunda más grande por número de estudiantes de la República del Ecuador; cuenta con 21 facultades, 66 carreras, 11 institutos y 4 centros, lo que implica una gran diversidad y complejidad en sus procesos académicos. Además, la UCE tiene como misión contribuir al desarrollo científico, tecnológico, cultural y social del país, mediante la formación integral de profesionales competentes, críticos y comprometidos con la sociedad. Por lo tanto, el análisis de los datos de los estudiantes de la UCE puede aportar información valiosa para mejorar la gestión académica, detectar posibles casos de error, fraude o corrupción, y garantizar el cumplimiento de los objetivos institucionales.

Las investigaciones realizadas en relación al aprendizaje de representación con autoencoders y las técnicas de agrupamiento serán aplicadas al análisis del progreso académico de los estudiantes universitarios a partir de sus registros personales.

El objetivo es aprender una transformación de los datos originales a un espacio de menor dimensión, que preserve o resalte las características relevantes y luego aplicar diferentes técnicas de agrupamiento para obtener grupos significativos y coherentes de estudiantes. El análisis de los datos de los estudiantes puede servir para identificar error, fraude y corrupción en los procesos del vicerrectorado académico en las instituciones de educación superior, ya que puede revelar patrones anómalos, inconsistencias o irregularidades en los procesos académicos y administrativos que afectan a la calidad y la equidad de la educación.

Referencias

1. Mohamed Nafuri, A.F.; Sani, N.S.; Zainudin, N.F.A.; Rahman, A.H.A.; Aliff, M.Clustering Analysis for Classifying Student Academic Performance in Higher Education. Appl. Sci. 2022, 12, 9467.
2. Valarmathy N., Krishnaveni, S. (2019) Performance Evaluation and Comparison of Clustering Algorithms used in Educational Data Mining. International Journal of Recent Technology and Engineering (IJRTE), 7 (655). ISSN 2277-3878.
3. Tschannen, M., Bachem, O., Lucic, M. (2018). Recent Advances in AutoencoderBased Representation Learning. arXiv preprint arXiv:1812.05069.
4. Huang, X., Hu, Z., Lin, L. (2023). Deep clustering based on embedded auto-encoder.Soft Computing, 27, 1075–1090.
5. Lu, S., Li, R. (2022). DAC–Deep Autoencoder-Based Clustering: A General DeepLearning Framework of Representation Learning. In: Arai, K. (eds) Intelligent Systems and Applications. IntelliSys 2021. Lecture Notes in Networks and Systems, vol 294. Springer, Cham.
6. Kanazawa, S., Sugiyama, Y., Yang, T., Goto, M. (2019). A Study of Feature Clustering Analysis based on the Hidden Layer Representation of an Autoencoder. Total Quality Science, 5(1), pp.1-10.
7. Jimbo-Santana, P., Lanzarini, L., Jimbo-Santana, M., Morales-Morales, M. (2023). Inteligencia artificial para analizar el rendimiento académico en instituciones de educación superior. Una revisión sistemática de la literatura. Cátedra, 6(2), pp.30–50.
8. Forest F., Lebbah M., Azzag H., Lacaille J. (2021). Deep embedded self-organizingmaps for joint representation learning and topology-preserving Clustering. Neural Computing and Applications, 33(24), pp.1-31

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Analysis of Bioinformatic algorithms for MSA

Adrián Díaz (student)[1] and Gabriela Minetti (director)[2] [0000-0003-1076-6766]

Universidad Nacional de Quilmes

Facultad de Ingeniería, Universidad Nacional de La Pampa, Argentina
diaz.adrian.g@gmail.com, minettig@ing.unlpam.edu.ar

Abstract. Aligning three or more biological sequences, such as DNA, RNA, or protein, is known as multiple sequence alignment (MSA). MSA is crucial in identifying important information about the sequences, including function, evolution, and structure. It serves as the first step in analyzing phylogenetic, protein, and genomic data. However, as sequence scale increases and the demand for alignment accuracy grows, MSA faces new challenges. Therefore, developing an efficient and precise tool for MSA and comparing its performance with existing ones has become a research hotspot in Bioinformatics. In this magister thesis, we propose a metaheuristic algorithm to solve MSA and a methodology to compare the performance of algorithms for aligning multiple sequences.

Keywords: Multiple sequence alignment, Bioinformatics, Simulated Annealing, Metaheuristics

1. Introduction

Bioinformatics is an interdisciplinary field that brings together experts in molecular biology, mathematicians, engineers, and physicists to align biological sequences, analyze genomic sequences, identify and predict molecular structures, determine the gene expression profile, etc. This discipline includes the computational techniques and applications that carry out these activities; being multiple sequence alignment (MSA) one of the most important bioniformatic problems [1].

Multiple sequence alignment is the process of aligning three or more biological sequences, typically DNA/RNA/protein. As the first step in phylogenetic, protein, and genomic analysis, MSA reveals the potential information about biological sequences, such as function, evolution, and structure. Obtaining an MSA is no trivial task because it involves complex calculations, and the results are not always biologically accurate. In addition, the required computational effort is highly dependent on the number of sequences to be aligned, becoming this kind of optimization in an NP-hard problem [1]. Although the execution speed and the result quality have improved over the years, producing alignments with insertions, deletions, and gaps in positions closer to biological reality is still a problem to be studied by bioinformatics researchers. As a consequence, the amount of software available to perform multiple alignment is large and diverse. Notice that, according to Google Scholar, in May 2023, the five most cited MSA software systems are: Clustal Omega [15], MUSCLE [6], KAlign [10], T-Coffe [5], and MAFFT [8].

Based on the above, our main goals are to design an algorithm that solves efficiently MSA and propose a well-defined methodology to compare different algorithms for MSA. At this research stage, we adapt an stochastic algorithm to align multiple sequences and introduce an algorithmic comparison methodology. A future step in this

magister thesis is to implement the suggested comparison method and to evaluate the performance of the proposed algorithm compared to these five algorithms on various biological sequences.

2. Multiple Sequence Alignment

Srinvias Aluru [1] defines an alignment of a set of sequences ($\bar{s_1}$, $\bar{s_2}$, ..., $\bar{s_N}$) as a correspondence among elements, $s'_{n_i}s$, one taken from each sequence, allowing for the absence of correspondents but keeping the order within each sequence. Where s_n is a string that represent a DNA or protein sequence; consequently, the s_n alphabets can be $\bar{\Sigma} = \{A, C, G, T\}$ or $\Sigma = \{the\ 20\ amino\ acids\}$, respectively. When $N = 2$, an alignment is called a pairwise sequence alignment (PSA), but if $N \geq 3$ is named a multiple sequence alignment (MSA). An alignment can be represented by a rectangular matrix $A = \{a_{nl}\}(1 \leq n \leq N, 1 \leq l \leq L)$ over $\Sigma' = \Sigma \cup \{-\}$, where a dash '-' denotes a null implying the absence of the correspondent. One or more contiguous nulls in a row is called a gap or an indel (insertion-deletion). As a long gap may be produced by a single evolutionary event, gaps and nulls refer to related yet distinct entities.

Given that the number of feasible alignments grows exponentially concerning the number of sequences and their length, the MSA objective is to find an alignment that maximizes its quality. The alignment quality is measured by an alignment score that generally takes the form of Equation 1, considering an MSA of N sequences (A), where $S_N(\mathbf{a}_l)$ indicates the similarity score assigned to the column vector $\mathbf{a}_l$[13,3,12], $G_N(*_l)$ refers the gap penalty given to column l, and the subscript N denotes the number of sequences involved.

$$H_N(\mathcal{A}) = \sum_{1 \leq l \leq L} \{S_N(\mathbf{a}_l) - G_N(*_l)\} \tag{1}$$

3. Simulated Annealing for MSA

Simulated Annealing (SA) [9] is an stochastic optimization algorithm that models the physical process of heating material. SA evolves by a sequence of changes between states generated by transition probabilities, which are calculated involving the current temperature. Therefore, SA can be modeled mathematically by Markov chains and avoids getting stuck prematurely at a local optimum by applying the Boltzmann probability. We select SA to solve MSA because this algorithm solved efficiently several NP-hard optimization problem in different real-world domains [7], including Bioinformatics [14]. To adapt SA to the MSA domain and obtain MSASA algorithm (MSA solved by SA), we represent an alignment as a state (or solution, S_i), define specific-domain move operators to generate a new state (S_j also called neighbor solution), and specify a scoring system to measure the state energy.

Alignment representation. To represent an alignment as a solution, we use a rectangular matrix $\bar{A}$ defined in Section 2, where the total number of sequences is N and the length of the longest sequence is L, including gaps. In Figure 1) five sequences, s_n, are aligned in a matrix A with $N = 5$ and $L = 9$.

	a_{n1}	a_{n2}	a_{n3}	a_{n4}	a_{n5}	a_{n6}	a_{n7}	a_{n8}	a_{n9}
$\overline{s_1}$	M	V	L	Y	C	D	-	-	-
$\overline{s_2}$	*V*	*C*	*D*	*E*	*F*	*Y*	*V*	*D*	*Y*
$\overline{s_3}$	*Y*	*F*	*A*	*A*	*D*	-	-	-	-
$\overline{s_4}$	*A*	*C*	*D*	*Y*	*C*	*V*	-	-	-
$\overline{s_5}$	*A*	*C*	*D*	*F*	-	-	-	-	-

Fig.1. Alignment of five sequences, A = $\{a_{nl}\}(1 \leq n \leq 5, 1 \leq l \leq 9)$.

InDelGap, a specific-domain move operator. InDelGap operator is based on the insertion operator for permutation representations and adapted to the MSA domain. InDelGap generates a new state from a previous one by inserting or deleting a gap in a sequence, s_n. At beginning, the operator uniformly selects either an insertion or deletion operation. Next, a sequence s_n is picked at random, which must contain at least one gap if the operation is a deletion. Then, a position is randomly chosen within s_n, and a gap is inserted or deleted at that location. Finally, the InDelGap operator ensures that all rows in A reach the new maximum length, L, by filling any shorter rows with gaps.

Scoring sytems. Measuring the state energy means evaluating the alignment quality through an MSA score, H_N(A). In this work, we compute the score using Equation 2, which sums the score calculated for each sequence pair. In this sense, we employ three different heuristics to compute the score for a sequence pair: maximum conservation (MC), substitution matrix (SM), and variability (Vb). MC penalizes the pairs of residuals that are not equal. In this way, MC tries to minimize the difference in residues per column and to propend to high conservation in each column. SM uses the *BLOSUM62* matrix to obtain the score between two residues. Consequently, biological context is incorporated into the alignment evaluation. Vb is Similar to MC but allows more variability per column since no penalty is applied when differences in pairs of residuals appear.

$$H_N(\mathcal{A}) = \sum_{1 \leq i \leq N} \sum_{1 \leq j \leq N} \{score(s_i, s_j)\} \tag{2}$$

4. Comparison methodology

To compare various algorithms that perform MSA in an automated and reproducible way, we propose a method that includes a set of tasks for executing each algorithm, utilizing test cases with varying characteristics, and applying tools to assess alignment quality, as is described in the following paragraphs.

Algorithm Selection. For selecting the algorithms for comparison in an unbiased way, we propose choosing the algorithms more cited in the literature. For this work, we use the five most cited MSA software systems, according to Google Scholar in May 2023: Clustal Omega, MUSCLE, KAlign, T-Coffe, and MAFFT.

Sequence Database Selection. To determine a representative database of test sequences, we also recommend using a benchmark widely accepted by the Bioinformatic community to evaluate and compare different MSA software. As a

consequence, we select BAli BASE (Benchmark Alignment dataBASE) [2] that contains high-quality, manually constructed multiple sequence alignments together with detailed annotations, whose alignments are all based on threedimensional structural superpositions and include linear motifs. Moreover, this database is organized into ten reference groups of sequences with different degrees of similarity and complexity.

Scenario Configuration. Configuring the test scenarios from the chosen sequence database is imperative to select the appropriate test cases. We develop the test scenarios based on three dimensions that are crucial in determining the accuracy and effectiveness of the research: *i)* sequence length, short (< 100 residues), medium (≥ 100 and < 400 residues), and long (≥ 400);*ii)* the number of sequences to align, low (< 100 sequences) and high (≥ 100 sequences); and *iii)* the identity percentage of the reference alignment, which is set in 30% as a general rule by the Bioinformatic community. Therefore, 23 test cases with diverse characteristics and complexity levels are used for this comparison.

Metric Selection. Selecting the metrics to compare the results obtained by each MSA software is essential to analyze their performance from different points of view. Because of this, we use metrics provided by different origins: *i) Sumof-pairs* (SP) and *Column Score* (CS) from BAli Score [2]; *ii)* percentages of identity and coverage from Mumsa [11]; and *iii)* the transitive consistency score (TCS) [5] from T-Coffee.

Execution Environment Configuration. To ensure the same execution environment in an algorithmic comparison, we configure and use virtual machines with the same initial hardware and software conditions. Furthermore, since we are working with stochastic algorithms or components (such as MSASA, Clustal Omega, and MAFFT), we need 30 independent runs for each test case and algorithmic configuration. The experimentation is carried out by using the NextFlow programming language, which provides a framework for creating workflows [4], as shown the Fig. 2, and allows to collect the results in a structured and automatic way, facilitating their subsequent analysis.

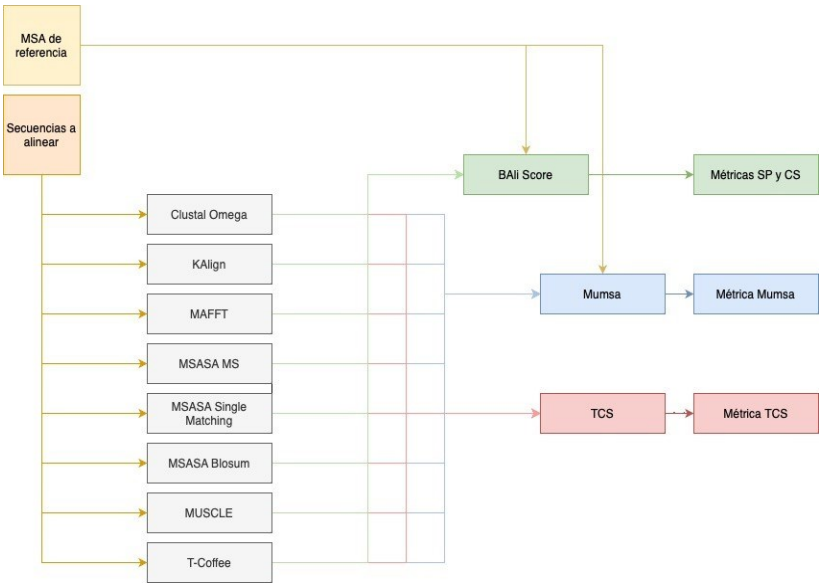


Fig.2. Experimentation Workflow.

5. Discussion and future works

In this research phase, we can study some preliminary results of the whole experimentation, as shown the Fig. 3. The CS metric values show that the five most cited MSA algorithms outperform our proposal, indicating that MSASA must be improved by tuning the move operator. Another future research work involves identifying the most appropriate MSA algorithm according to the characteristics of sequences to align.

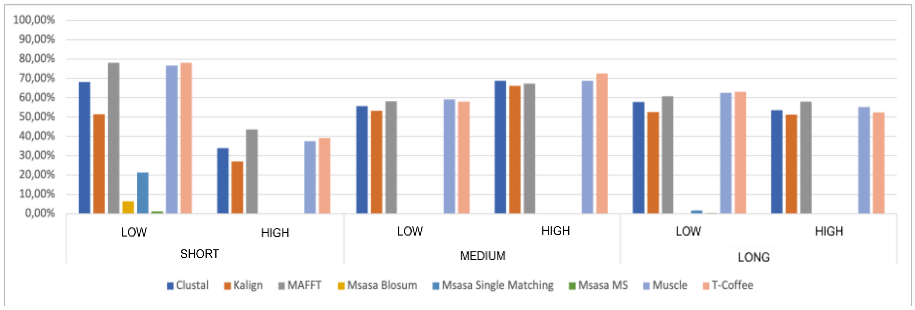


Fig.3. CS metrics values for the results obtained by each MSA algorithm grouped by number of sequences and sequence lengths.

References

1. Aluru, S.: Handbook of Computational Molecular Biology (Chapman & All/Crc Computer and Information Science Series). Chapman Hall/CRC (2005)
2. Bahr, A., Thompson, J., Thierry, J., Poch", O.: Balibase (benchmark alignment database): enhancements for repeats, transmembrane sequences and circular permutations. Nucleic Acids Res. 29, "323–326" (2001)
3. Carrillo, H., Lipman, D.J.: The multiple sequence alignment problem in biology. SIAM Journal on Applied Mathematics 48(5), 1073–1082 (1988)
4. Di Tommaso, P., Chatzou, M., Floden, E.W., Barja, P., Palumbo, E., Notredame, C.: Nextflow enables reproducible computational workflows. Nature Biotechnology 35, 316–319 (04 2017)
5. Di Tommaso, P., Moretti, S., Xenarios, I., Orobitg, M., Montanyola, A., Chang, J.M., Taly, J.F., Notredame, C.: T-coffee: a web server for the multiple sequence alignment of protein and rna sequences using structural information and homology extension. Nucleic acids research 39(Web Server issue), W13–W17 (2011)
6. Edgar, R.C.: Muscle: multiple sequence alignment with high accuracy and high throughput. Nucleic acids research 32(5), 1792–1797 (2004)
7. Hernández, J., Salto, C., Minetti, G., Carnero, M., Sánchez, M.: Hybrid simulated annealing for optimal cost instrumentation in chemical plants. Chem. Eng. Trans. 74, 709–714 (2019)
8. Katoh, K., Standley, D.M.: MAFFT multiple sequence alignment software version 7: improvements in performance and usability. Molecular Biology and Evolution 30(4), 772–780 (2013)
9. Kirkpatrick, S., Jr, C.G., Vecchi, M.: Optimization by simulated annealing. Science (220), 671–680 (1983)
10. Lassmann, T.: Kalign 3: multiple sequence alignment of large data sets. Bioinformatics 36(6), 1928–1929 (2019), advance online publication
11. Lassmann, T., Sonnhammer, E.: Kalign, kalignvu and mumsa: Web servers for multiple sequence alignment. Nucleic acids research 34, W596–9 (08 2006)
12. Pevzner, P.: Computational molecular biology: An algorithmic approach. The MIT Press (2000)
13. Sankoff, D., Cedergren, R.J.: Simultaneous comparison of three or more sequences related by a tree. In: Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison. Addison-Wesley, Reading, MA (1983)
14. Sarıyer, O.S., Güven, C.: Sequence alignment using simulated annealing. Physica A: Statistical Mechanics and its Applications 389(15), 3007–3012 (2010)
15. Sievers, F., Higgins, D.G.: Clustal omega, accurate alignment of very large numbers of sequences. Methods in Molecular Biology (Clifton, N.J.) 1079, 105–116 (2014)

Coordinadores:

- Rodolfo Bertone (UNLP)
- Hugo Alfonso (UNLPam)
- Nora Reyes (UNSL)

CACiC 2023

CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

XXI Workshop de Bases de Datos y Minería de Datos

> Full Papers

WBDMD

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Descubrimiento de patrones relevantes del Dengue mediante la Programación Lógica Inductiva y *Twitter*

Carlos Alberto Benítez Galván, Tania Mabel Leguizamón Ovelar, Gustavo Sosa-Cabrera, and María E. García-Díaz

Universidad Nacional de Asunción - Facultad Politécnica, Paraguay
{carlosbenitez, tleguizamon}@est.pol.una.py, {gdsosa, mgarcia}@pol.una.py
http://www.pol.una.py

Resumen El dengue es una infección vírica transmitida por la picadura de las hembras infectadas de mosquitos del género Aedes Aegypti. Es una enfermedad infecciosa transmitida por mosquitos, una de las principales causas de enfermedad y muerte en las regiones tropicales y subtropicales, incluido Paraguay. El dengue, en nuestro país, ya es considerada una enfermedad endémica, es un problema cotidiano de salud pública desde el año 2009, en cuyo año la epidemia se instaló sobre territorios endémicos con circulación simultánea afectando a un gran número de la población y trasladando susceptibilidad a población de niños y adolescentes. En este artículo analizamos cómo se refleja la epidemia de dengue en *Twitter*, el cual, es un canal de redes sociales único, en el sentido de que los usuarios discuten y hablan sobre los temas más diversos, incluidas sus condiciones de salud. Se analizaron los datos pertenecientes a un período de tiempo sobre el fenómeno que nos ocupa, teniendo en cuenta que, en la era digital, las redes sociales son un instrumento clave para informar a la población sobre cualquier emergencia sanitaria y utilizar ese mismo conocimiento para medir el grado en que la misma puede utilizarse en aras de la vigilancia.

Keywords: Dengue · Redes Sociales · Twitter · Vigilancia · Salud · Paraguay.

1. Introducción

Muchas de las enfermedades emergentes que amenazan la salud pública mundial son transmitidas por vectores artrópodos y constituyen un problema global cuyo impacto social y económico es enorme. Entre los principales vectores de enfermedades se encuentran los mosquitos, los cuales son capaces de transmitir el dengue y otras enfermedades. Con respecto a lo anterior, el dengue es una enfermedad infecciosa, la cual sin tratamiento médico, en algunos casos, puede llevar a la muerte. Se transmite mediante la picadura de la hembra del mosquito infectado Aedes Aegypti, principal vector, que se cría en el agua acumulada de recipientes y objetos en desuso [1]. Actualmente la enfermedad se ha extendido por gran parte de las regiones de clima tropical y ecuatorial del mundo.

Según los reportes del Centro Provincial de Información de Ciencias Médicas [2] en las últimas décadas ha aumentado enormemente la incidencia de dengue en el mundo. Una gran mayoría de los casos son asintomáticos, por lo que el número real de casos está insuficientemente notificado y muchos están mal clasificados. En Paraguay, la enfermedad ya está instalada de forma continua y constante [3] y si bien el dengue ya tuvo sus brotes en las zonas urbanas, Paraguay no presentaba víctimas mortales por dengue hasta el año 2007, resaltando que en el país se vivieron dos epidemias de gran

magnitud: la de 2013 y la de 2016; siendo el peor registrado en la historia del país la epidemia ocurrida en el año 2013.[4]

Por lo tanto, en el campo de la salud pública, el internet cambió la vigilancia de la salud mundial, al reducir drásticamente el tiempo que se requiere para reconocer los brotes y apoyar a la distribución mundial de información de salud. Cuando aparecieron los sistemas en internet en el escenario de la salud pública, se les crítico por la posibilidad de información falsa, por la falta de especificación y su sensibilidad. Aunque se reconocen estos problemas y se toman en cuenta lo mejor posible, las críticas siguen siendo válidas. Pero a pesar de que existan posibilidades de una equivocación, los sistemas de información de epidemias en internet son una fuente invaluable de información acerca del surgimiento de enfermedades.

En la perspectiva de analizar los datos o informaciones extraídas de las redes sociales para predecir o anticipar eventos futuros, se propone insertar un sistema de aprendizaje automático para dicha enfermedad, enfocado en la utilización de medios sociales, en donde, como principal fuente de observaciones se propone la utilización de la red social *Twitter*, acompañado de un método de pre-procesamiento basado en la matemática de primer orden y el lenguaje natural. Los mismos servirán como datos de entrada para un algoritmo de primer orden denominado FOIL, uno de los sistemas más conocidos dentro de la Programación Lógica Inductiva. Dicho sistema proporcionará un espacio para la circulación de información y conocimiento de carácter estratégico, a través del aprendizaje de reglas, que permitirá la toma de decisiones, centrándose especialmente en la forma que los medios sociales pueden ser movilizados para establecer una relación entre la ciencia y la salud.

2. Materiales y Métodos

Para expresar conocimiento sobre situaciones del mundo real o situaciones de nuestro interés hacemos uso de enunciados declarativos. Decimos que son enunciados declarativos porque se trata de expresiones del lenguaje natural que bien pueden ser verdaderas o falsas. Tomando esto como punto de partida se analizaron a continuación, un conjunto de enunciados declarativos que fueron extraídos de una fuente de datos, fueron evaluados y posteriormente introducidos al algoritmo de FOIL.

Los datos en cuestión provienen de la red social *Twitter*. Recordemos que *Twitter* es una red social en auge en nuestro país y sobre todo que, como plataforma de comunicación, *Twitter* se ha infundido cada vez más en la vida diaria, independientemente de su ubicación geográfica. Datos estimativos, reportan que existen 500 millones *tweets* enviados cada día, eso es 6.000 *tweets* cada segundo.[5]

Teniendo en cuenta lo anterior, haremos uso de la red social *Twitter* como fuente de datos para llevar a cabo nuestra investigación y comprobar que tan útiles son los datos extraídos del mismo para detectar una epidemia de dengue en el Paraguay, específicamente de la ciudad Asunción.

Los algoritmos hechos, tanto para la captura y el procesamiento de textos fueron escritos en el lenguaje de programación Python, debido a su flexibilidad, sencillez y facilidad en la escritura. Además de todo esto, su código es más organizado comparando con otros lenguajes de programación.

2.1 Captura de datos.

Fueron recolectados 500.000.000 *tweets*, de un periodo de tiempo que abarca desde Setiembre 2017 hasta Febrero 2018, correspondientes a 6 meses antes de ocurrida la epidemia registrada por el Ministerio de Salud Pública y Bienestar Social junto con la Dirección Nacional de Vigilancia Sanitaria. En sus reportes indican que en el año 2018, así como se muestra en la Figura 1, desde la semana 1 se registró un aumento del número de notificaciones de arbovirosis, a mediados del mes de marzo (semana 11) se reportó el mayor número de casos notificados; entre las semanas 30 y 41 se notificaron menos de 100 sospechas de arbovirosis (periodo interepidémico). A partir de la semana 42 se registraron nuevamente más de 100 sospechas de arbovirosis, y entre las semanas 45 y 52 se registraron más de 200 notificaciones por semana a nivel nacional.[6]

Para la recolección de los *tweets* se usó una librería llamada *Get Old Tweets*, en su versión 3, el cual nos permite capturar *tweets* históricos de más de siete días de duración, ya que no tiene en cuenta la limitación del tiempo.

2.2 Pre-procesamiento de datos.

Para la recolección de los datos, se definió un terminó de búsqueda por el cual se hará la extracción. En este caso se tomaron las menciones de autoridades y medios de prensa en donde se reportan los reclamos hechos por la ciudadanía referentes a las malas condiciones relacionados a la provisión de servicios públicos e infraestructura (electricidad, agua potable, recolección de basura, reportes de baches, etc.)

Una vez recolectados todos los *tweets*, es necesario procesarlos para reunir sólo aquellos *tweets* que hagan referencia a la enfermedad o que contengan palabras claves, también llamados patrones, que hagan alusión a una posible aparición de la enfermedad. Se definió entonces dentro del algoritmo un filtro de expresiones regulares para el procesamiento de los mismos.

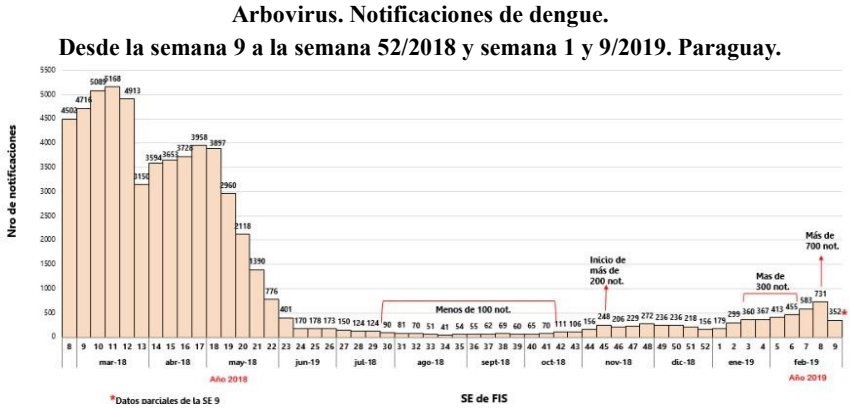


Figura1. Fuente: Dirección Nacional de Vigilancia Sanitaria – Paraguay

Esas expresiones regulares a buscar dentro de cada uno de los textos son: lluvia, agua servida, mosquitos, criaderos, basuras, desechos, inundaciones, dengue, etc.

Un punto importante a mencionar, es que sólo 138 *tweets* contenían esas palabras claves, por lo que nuestro conjunto de datos se redujo bastante.

Posteriormente los *tweets* fueron filtrados y separados en pequeñas unidades gramaticales utilizando una librería de procesamiento de lenguaje natural (NLP), denominada *SpaCy*, cuya funcionalidad incluyen una herramienta denominada *Part-Of-Speech tagging* (la cual permite etiquetar las palabras como una de varias categorías para identificar la función de la palabra en un idioma determinado. Las categorías de parte del discurso incluyen sustantivo, verbo, artículo, adjetivo, preposición, pronombre, adverbio, conjunción).

Luego de realizar esto, se procedió a implementar nuevamente un algoritmo que se encargaba de filtrar y extraer solo aquellos sustantivos detectados por el analizador sintáctico. Se tuvo en cuenta sólo los sustantivos, debido a que, los mismos pertenecen a la definición de constantes dentro de la Lógica de Primer Orden, es decir, representan los objetos del dominio de datos. Con esos sustantivos se realizó una serie de combinaciones entre sí, con lo que se ha dado inicio al diseño de la estructura de la base de datos lógica.

Si bien, muchas de esas combinaciones de sustantivos no tenían sentido, es importante analizar la estructura del *tweet*. Por ejemplo, dada la siguiente combinación.

(compromiso, sector).

Veamos la estructura del *tweet*:

#Dengue “El compromiso del sector privado no solo fue retorico, sino también práctico. También las FF.AA. que nos permitió acceder a herramientas que no solemos contar” @Ferreiromario1 Vía @edugimenez93 780AM.

En ese caso, se hace mención a la enfermedad a través del #Dengue, con ese *tweet* se puede entender que la enfermedad no solo es un problema del sector público, sino también afecta al sector privado, el cual provee herramientas nuevas para combatir la enfermedad.

Las distintas combinaciones de sustantivos generadas serían los ejemplos positivos que necesitaría FOIL para realizar el análisis. Recordemos que FOIL es un algoritmo que aprende un conjunto de reglas de primer orden, para explicar un predicado objetivo P, a partir de un conjunto de ejemplos positivos y predicados de soporte. Para ello, usa una estrategia muy similar a la de los algoritmos de cobertura secuencial donde aprende una regla en cada iteración, eliminando las tuplas positivas del conjunto de entrenamiento cubiertas y repitiendo el mismo proceso, hasta que no queden tuplas positivas por cubrir.[7]

En su estrategia de funcionamiento, FOIL parte de la regla más general con el predicado objetivo como consecuente y sin antecedente y le va añadiendo los predicados a su antecedente que considera óptimos en el momento de su elección, según una técnica de decisión heurística que explora entre unos tipos preestablecidos de predicados. De esta forma prosigue hasta obtener una regla que cubra solamente ejemplos positivos[8].

La Ganancia FOIL(R,L) está dada por t:

$$\left(log_2 \frac{p_1}{p_1 + n_1} - log_2 \frac{p_0}{p_0 + n_0} \right)$$

Donde:

- **R** es la regla en construcción en su fase inmediatamente previa a añadir el nuevo literal que salga elegido en el proceso.
- **L** es el literal que se valora para su incorporación en la regla.
- **t** es el número de instancias positivas cubiertas por la regla R antes de añadir L y que genera al menos una extensión positiva cubierta por la regla al añadir L a su antecedente.
- p_0 es el número de tuplas positivas cubiertas por la regla R antes de su incorporación de L. n_0 es el número de tuplas negativas cubiertas por la regla R antes de su incorporación de L.
- p_1 es el número de tuplas positivas cubiertas por la regla R tras su incorporación de L.
- n_1 es el número de tuplas negativas cubiertas por la regla R tras la incorporación de L.

Al igual que los algoritmos de recubrimiento, tras eliminar los ejemplos cubiertos por la regla recién construida, continúa el cálculo de la siguiente a partir de los ejemplos positivos remanentes. Así hasta alcanzar una familia de reglas que cubra solo a los ejemplos positivos. Los ejemplos negativos se clasifican por defecto. [8]

La elección de uno de los literales entre todos los candidatos se realiza mediante una optimización que se basa en la medida de la información de Shannon. Concretamente, se utiliza la Ganancia-FOIL. Ver figura 2. [9][8]

Por lo tanto, el algoritmo necesita de ejemplos positivos, ejemplos negativos y conocimiento base para llevar a cabo el análisis y descubrimiento de las posibles reglas. En cuanto a los ejemplos negativos, el usuario puede indicarle a FOIL que genere por sí mismo los ejemplos negativos mediante la Hipótesis del Mundo Cerrado, ´esta asume que toda la información positiva relevante ha sido especificada; cualquier otro hecho no especificado se asume por defecto que es falso. Por lo que para el análisis de este trabajo se deja que FOIL genere por si mismo estos ejemplos. El otro punto respecto al conocimiento base, hace referencia a indicarle a FOIL que son cada uno de los objetos pertenecientes al análisis, en este caso, son sustantivos que hacen mención al dengue, por lo que se declara los mismos dentro del algoritmo y así se genera la base de datos lógica completa. A continuación se muestra un diagrama explicativo, donde se detalla todo el proceso llevado a cabo para la generación de la base de datos lógica completa.

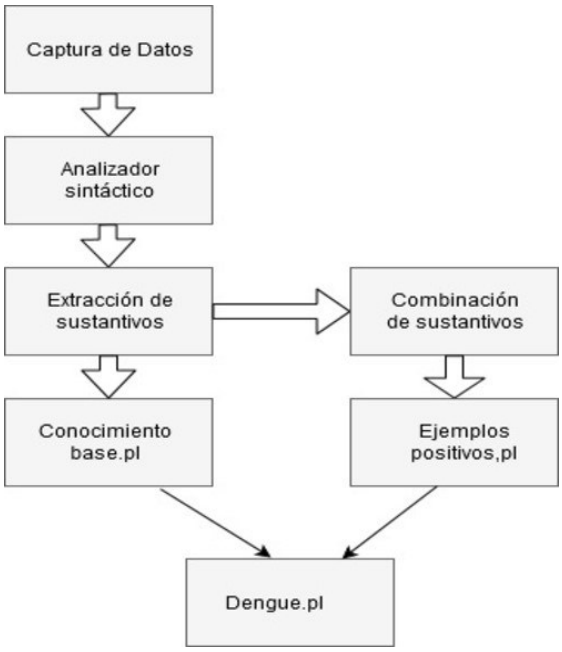


Figura2. Proceso de creación de la base de datos lógica

3. Resultados y Discusión

Teniendo lista la base de datos con la extensión .pl, extensión correspondiente al lenguaje PROLOG, con la estructura y las declaraciones que requiere el algoritmo FOIL, se procede a ejecutarlo.

En primer lugar, se abre una consola de comandos en GNU/LINUX o en el sistema operativo que se utilice, en donde se carga primeramente el algoritmo a través de la siguiente instrucción.

1. [foil].

Seguidamente se ejecuta la base de datos lógica creada.

2. [dengue].

Al final se invoca al algoritmo FOIL y se pasa como parámetro el predicado objetivo P que se quiere conseguir a través del mismo, con su aridad correspondiente. En este caso se busca una relación entre las combinaciones hechas con los *tweets* extraídos. Recordar que la aridad del mismo sería dos, debido a que las combinaciones fueron hechas en parejas de dos.

3. foil (aparece tweet/2).

Al ejecutar en la base de datos, con todos los ejemplos positivos generados y el conocimiento base declarado, FOIL fue incapaz de encontrar alguna regla. Esto se debe a una razón muy importante: el algoritmo en su búsqueda no encontraba una relación entre todos los predicados ya declarados, por lo que se procedió a buscar una estrategia que permitiera que esas reglas aparezcan.

Según el Ministerio de Salud Pública y Bienestar Social del Paraguay los mosquitos transmisores de la enfermedad se crían en neumáticos viejos, floreros, tanques, tambores y recipientes que puedan contener agua[10]. El Ministerio de Salud de Colombia concuerda con lo mismo, mencionando que los mosquitos ponen sus huevos en depósitos de agua limpia como albercas, floreros de plantas acuáticas, llantas, baldes de agua y cualquier recipiente que está a la intemperie y que puede almacenar agua[11].

Esto ha planteado a realizar una serie de cuestionamientos, principalmente al plantear una estrategia teniendo en cuenta los factores que deben ocurrir para afirmar que pueda existir un posible criadero o una epidemia en un sector. A continuación, se generaron esos factores como datos sintéticos y fueron insertados como conocimiento base dentro del algoritmo.

Se volvió a ejecutar FOIL en búsqueda de una regla valida y esta vez el mismo encontró tres reglas que se muestran a continuación:

1. aparecetweet(B,A) :- [factor genera(A,B),factor genera(A,C)], aparece tweet(C,A).
2. aparecetweet(A,C) :- [factor genera(B,A), factor genera(B,C)], aparece tweet(A,B).
3. aparecetweet(B,A) :- [factor genera(A,C), factor genera(B,C)], aparece tweet(C,A).

Se debe mencionar que, la palabra aparece tweet fue utilizada como nexo entre las distintas combinaciones que fueron hechas para los ejemplos positivos y la palabra factor genera para las declaraciones dentro del conocimiento base.

3.1 Demostración de reglas

Se toma una de las tres reglas encontradas por FOIL para demostrar la validez de la misma y la relación existente entre (B,A) dentro del mismo conjunto de datos.

Regla descubierta por FOIL:

[factor genera(A,C) ˆ factor genera(B,C)] ˆ aparece tweet(C,A)

Para realizar la demostración, se tuvieron en cuenta las propiedades de la Inferencia Lógica, que se conceptualiza como sigue:

“La Inferencia Lógica es una parte importante de la Lógica Maten ática y se puede expresar de la siguiente manera: De premisas verdaderas se obtienen sólo conclusiones verdaderas.“ [12]

Entonces:

- Dado que A y B sean las premisas que hacen referencia a los factores de dengue obtenidos por las distintas combinaciones hechas,
- Además que C sea una factor de dengue generado de manera sintética.

Se puede encontrar que aplicando la propiedad de simplificación a la regla se puede obtener lo siguiente:

factor genera(A,C) ˆ factor genera(B,C)

La propiedad de simplificación permite pasar de una conjunción a cada una de las proposiciones que están unidas por ” ˆ ” [12]

Si se observa bien, se nota que en ambos predicados existe el factor C, por lo que se procede a utilizar la propiedad de Igualdad Maten ática que es denotada de la siguiente manera:

"Si dos expresiones que hacen referencia al mismo objeto, entonces se puede asumir que existe una relación entre ambas expresiones."[13] Por lo que la regla, quedaría de la siguiente forma:

relación(A,B) = relación(B,A)

Teniendo en cuenta ésto, se puede decir que esas tres reglas se pueden tomar como una sola, debido que en ellas aparece un factor común referenciado por la letra C. La misma representa el nexo que debe existir entre los dos factores sobrantes A y B, así también representa el factor desencadenante que propicia la aparición de los demás factores, los cuales podrían generar un posible criadero y posteriormente una epidemia en un sector.

Si llevamos esto mismo, dentro de nuestro conjunto de datos, tendríamos lo siguiente:

Para el primer caso tomado del conjunto de datos:

factor _genera(agua servida,criaderos).
aparece tweet(agua servida,basuras).

Según la regla descubierta por FOIL, en el cual C desencadena otros factores como A y B, es posible observar que en las expresiones anteriores el factor común en ellas sería la palabra "agua servida"; por lo que la misma desencadena la aparición de criaderos de mosquitos y basuras. Así también, dentro del mismo dominio de datos debe existir una combinación entre A y B. Veamos otro ejemplo similar.

factor genera(yuyales, focos).
aparece tweet(yuyales, propagación).

De la misma forma se entiende que, los yuyales en un sector propician focos de la enfermedad y permiten la propagación del mismo.

4. Conclusión

Con todo este proceso que se llevó a cabo, desde extraer textos, documentos o registros no estructurados de una fuente de datos, separar la información en pequeñas unidades gramaticales, que para este caso en particular sólo se necesitó las constantes encontradas en una sentencia, las cuales forman parte de la matemática de primer orden para finalmente someterlas a la técnica de Programación Lógica Inductiva utilizando un algoritmo conocido como FOIL; se pudo generar reglas, a partir de las cuales podemos establecer criterios y conclusiones comprobadas en la sección 3.1.

El algoritmo en estudio demostró ser capaz y eficiente para encontrar reglas dado un tema específico, teniendo en cuenta la estructura adecuada y además del conocimiento *a priori* que debe ser insertada dentro del mismo, para que de este modo, pueda encontrar las relaciones que se deben establecer dentro del conjunto de datos que permitan arrojar las reglas correctas.

FOIL uno de los algoritmos más simples dentro de la técnica utilizada, puede ser insertado en un futuro dentro de un sistema de vigilancia sanitaria para ayudar a determinar los factores determinantes que propician la aparición de las enfermedades y que podría llevar a una posible epidemia sino se tienen en cuenta las acciones de salubridad respectivos.

Referencias

1. MSc. Miguel Lugones Botell Dra. Marieta Ramírez Bermúdez. Dengue. Revista Cubana de Medicina General Integral, 2012.
2. Infomed Matanzas. Dengue: de acuerdo con la OMS, uno de los diez desafíos prioritarios en materia de salud para 2019. Centro Provincial de Información de Ciencias Médicas, Setiembre 2019.
3. Organización Panamericana de la Salud. Dengue: Es preocupante que la prevalencia mundial se haya incrementado dramáticamente en la última década. https://www.paho.org/par/index.php?option=com_content&view=article&id=442:
4. dengue-preocupante-que-prevalencia-mundial-haya-incrementado-dramaticamente-ultima-decada& Itemid=258. Último acceso 11/11/19.
5. Celsa Pereira. Sin epidemia de dengue tras 14 años. ABC Color Paraguay, Marzo 2014.
6. Mohamed Sikandar G. 100 social media statistics you must know 2018. https:// blog.statusbrew.com/social-media-statistics-2018-for-business/. Ultimo´ acceso 10/10/19.
7. Dirección General de Vigilancia Sanitaria. Boletín epidemiológico semanal. http://vigisalud.gov.py/webdgvs/files/boletines/SE4_2019_Boletin.pdf. Último acceso 20/12/19.
8. Humberto Chaviano Arteaga. La Programación Lógica Inductiva para el Aprendizaje Automatizado de conceptos. 2011.
9. Enseñanza y aprendizaje asistido por computador. di002.edv.uniovi.es/~alguero/eaac/old/2.4apuntesaprendizaje99.ps. Último acceso 20/11/19.
10. Federico Holik. Teoría de la información de Claude E. Shannon. http://dia.austral.edu.ar/Teor%C3%ADa_de_la_informaci%C3%B3n_de_Claude_E._Shannon. Último acceso 13/01/20.
11. Lic. Sara Constatini Lic. Edgar Sanabria. Programa nacional de control vectorial del dengue. SENEPA Paraguay, 2018.
12. Ministerio de Salud y Protección Social. Dengue. https://www.minsalud.gov.co/salud/publica/PET/Paginas/dengue.aspx. Último acceso 19/10/19.
13. P. Suppes & S. Hill. Introducción a la lógica matemática. 1963.
14. Guillermo Restrepo. Fundamentos de las matemáticas. 2003.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Performance analysis of the Survival-SVM classifier applied to gene-expression databases

Genaro Camele[1] and Waldo Hasperué[1,2]

Instituto de Investigación en Informática (III-LIDI), Facultad de Informática
Universidad Nacional de La Plata.
Comisión de Investigaciones Científicas (CIC-PBA).
{gcamele, whasperue}@lidi.info.unlp.edu.ar

Abstract. The analysis of epigenetic information for the diagnosis and prognosis of patients has been gaining relevance in recent years due to the technological progress that entails a decrease in information extraction and processing costs. One of the tasks most commonly carried out in this area is obtaining models that allow using patient epigenetic information to make inferences about survival analysis. As a result, optimizing these models turns into a problem of great interest today. In this article, the evaluation of different metrics and execution times for the Survival Support Vector Machines model is carried out through survival analysis applied to gene expression databases. Different experiments were performed varying the number of genes used for training to measure the correlation between model performance and data growth. The results showed that linear and polynomial kernels offer a better balance between execution time and model predictive power when the number of genes to be evaluated is less than 2000, while the cosine and RBF kernels are better candidates otherwise.

Keywords: Survival analysis, Survival Support Vector Machines, Regression, Performance, Apache Spark.

1. Introduction

Functional genomics is responsible for the systematic collection of information on the function and interaction between genes and/or proteins. It allows an understanding of how the genome works as a whole, through the controlled expression of each one of its genes. In this area, gene expression profile analyses stand out; their main objective is identifying a group of genes whose expression pattern is associated with a specific phenotype, a concept known as gene expression signature, or simply signature. In medicine, signatures are particularly useful as a diagnostic, prognostic, or predictive biomarker of any given pathology. Biomarkers with prognostic value allow better stratification of patients according to their prognosis of disease progression regardless of a therapy, which opens the way to researching appropriate treatments for each defined patient category. The estimate of a biomarker's prognostic value is measured using survival analysis techniques.

Survival analysis consists of a series of statistical procedures for data analysis that study the time until an event occurs (such as patient death or tumor recurrence). The number of these procedures is large, and with the lower costs for data storage and processing and the technological advances in sequencing, there has been an exponential growth of data volumes, thus increasing the need for algorithms that meet the demand in an acceptable response time.

1.1 Motivation

The main reason for measuring the execution times of classifiers with survival data is their usefulness in Feature Selection (FS) algorithms for survival data. These algorithms are based on a gene expression dataset with prognostic, diagnostic, or predictive power in any pathology of interest. The objective of the FS is to obtain a subset of genes or molecules that have the same or better performance for the current problem. Due to the high computational cost that would require evaluating the entire solution space (i.e., all the combinations between all genes or molecules), population metaheuristics such as Binary Particle Swarm Optimization (BPSO) [1] or Binary Black Hole Algorithm (BBHA) [2] are used. These techniques intelligently traverse only a portion of the solution space until finding the subset that optimizes the fitness function, requiring much less time (knowing that an approximate solution is more likely to be found than the best solution).

The main problem is that the fitness function used by these metaheuristics involves the execution of classifiers that could require high computational power. For this reason, the parallelization and distribution of the classifier evaluation are essential to speed up the total execution time for the FS algorithm.

Multiomix [3] is a platform which allows researchers to upload datasets or use a public database (already integrated into the platform) to perform survival analysis. The development and measurement of classifiers such as Survival-SVM or Random Survival Forest [4] in distributed environments will allow the developer team to put these algorithms into production for free use by the community. Measuring the execution times for the different database models and size configurations allows defining a task distribution policy throughout the computer cluster (orchestrated by Apache Spark) to carry out the various experiments more effectively and efficiently. These optimizations are notably relevant for the community as several studies are available, including selecting the most appropriate model within a distributed environment, as in [5] where classifier models provided by Apache Spark MLib applied to FS techniques are compared. Or in [6], where models such as SurvivalSVM and Random Survival Forest are proposed to obtain a better C-Index than the Cox Proportional Hazards (CPH) offered by standard models for survival analysis in oncology studies. There are also algorithms focused on optimizing the training stage of the model without affecting the prognostic power against real data sets [7].

1.2 Survival Support Vector Machines

The Survival Support Vector Machines model [8][9] is an extension of the standard support vector machine for time and event data, with support for right-censored data, i.e., samples that do not present the event during a survival study. Its main advantage is that it considers complex and non-linear relationships between traits and survival. The kernel function of this model implicitly maps input features into highdimensional feature spaces that describe survival data with a hyperplane. That makes SVMs very versatile and applicable to a wide range of data.

Survival analysis in the context of support vector machines can be described in two different ways:

- As a ranking problem: the model learns to assign samples with shorter survival times a lower rank by considering all possible pairs of samples in the training data.
- As a regression problem: the model learns to directly predict log survival time.

The training data used to obtain such a model consists of a triplet (x_i, y_i, δ_i) where x_i is a d-dimensional feature vector, $y_i > 0$ is the survival or censoring time, and $\delta_i \in \{0, 1\}$ is the occurrence indicator for the event (0 = the event did not occur as is said to be censored, 1 = the event occurred). The model's objective will be to minimize the function presented in [8].

In this article, the experiments are limited to regression tasks only.

2. Experimentation

2.1 Hardware configuration and experiments

All experiments were carried out in an Apache Spark cluster made up of a single master node and three worker nodes. All four nodes had Ubuntu 20.04 LTS, an Intel(R) Core(TM) i3-4160 CPU @ 3.60GHz, and 8GB of RAM. As regards software, Spark 3.1.1 was used.

To avoid any bias in the results, a Cross Validation (CV) of 10 folds was performed. To obtain these folds, stratified sampling was used [10]. Additionally, the entire process of randomly selecting attributes, dividing into folds, training, and evaluating the models was performed 30 times with each type of kernel and optimizer available in the Scikit-Survival Python library. The kernels available for the SSVM algorithm are Linear, Cosine, Polynomial, and Radial Basis Function (RBF). As regards the optimizers, the library allows choosing between AVLTree and RBTree. The metric to be optimized is Harrell's concordance index (C-Index) [11][12], which is a suitable metric for survival analysis because it evaluates the predictive ability of models in terms of the relative order of survival times, considering both observed and censored events. By encompassing categorical and continuous data, and having a range of interpretable values between 0.5 and 1.0, the C-index allows comparing different models and selecting the most suitable one to predict events of interest over time, which makes it a versatile and widely accepted tool in medical and scientific research in general.

2.2 Algorithm and dataset used

The algorithm used to carry out the experiments is detailed below:

```
program SurvivalSVMExecution
parameters
    ds, {dataset to evaluate}
    n_individuals, {number of individuals}
begin
    total_n_f= get_dataset_n_features(ds)
    current_n_f = 10
    step = 100
```

```
    for iter in 1 .. 30 do
        while current_n_f <= total_n_f do
            for ind_idx in n_individuals do
                rand_features = get_rand_feat(ds,current_n_f)
                indivs[ind_idx] = rand_features
                if current_n_f > 100
                    current_n_f += step
                else
                    current_n_f = current_n_f + 10
            end
            results = compute_in_spark(indivs)
            store_results_in_csv(results)
        end
    end
end.
```

The dataset used for the experiments is the Breast Invasive Carcinoma (TCGA, PanCancer Atlas), which contains data from 1082 patients (931 samples without event occurrence and 151 samples with event definition) and 20014 genes [13].

As previously mentioned, 30 iterations are performed to achieve statistical significance. In each of them, *N* individuals (*indivs* variable) of the metaheuristic under study are computed (particles in the case of the BPSO algorithm, or stars in BBHA). Each of these individuals represents a random subset of features and computes prognostic power and other metrics from the assigned subset by the 10-folds CV.

The function *compute_in_spark* distributes the *N* individuals among the Workers of the Apache Spark cluster, where a Worker can receive several individuals with different subsets of genes to compute. Even so, the CV task is carried out on a single individual at a time to avoid variations in the time measurements and avoid bias when obtaining the real execution times with the parameters set.

The result of all computed individuals is expressed in the algorithm as results. This variable, where *N* is the number of individuals to be evaluated, consists of an *N*-tuple of the following data: the average fitness value (C-Index) for the every of the folds of the CV during the model training and validation phases, the time it took the Spark worker to compute a given agent (i.e., the entire CV process), the number of features evaluated by that worker, the time that took each Survival-SVM iteration, average testing time, number of iterations it took the SVM to converge, and other trace data within the cluster such as partition number within the Spark cluster and the hostname of the computer where the Spark Worker is operating.

3. Result and discussion

The results obtained after running the different gene subsets for 30 independent runs are discussed below.

Regarding the optimizers, the experiments were carried out with the AVLTree and RBTree structures. Both structures were found to impact required time for each iteration only during the training stage, thus not interfering with model fitness function. However, RBTree results were discarded because they solely led to a degradation in

training performance; for that reason, only AVLTree results were presented from now on.

As expected, the relationship between testing time and the number of features follows a linear trend (Fig. 1) since there is no error function to optimize in the process, and the computation of the inference only consists of the execution of the Survival-SVM kernel function.

The direct relationship between run time and number of features presents an abrupt change after 100 genes as very high run times are observed for the CV process evaluating a reduced set of genes. After that number the relationship starts to be more linear in nature with the Linear, Polynomial kernels, while in the case of the RBF and Cosine kernels, the difference is significant. The Cosine kernel presents an erratic runtime with high variance, stabilizing after 2500 genes, while RBF flattens out after 100 genes and varies with increasing features. All these details can be seen in Fig. 2.

Based on the total CV time, individual iteration times also vary for the same kernels that are inefficient with few genes (Fig. 3). However, for the Linear and Polynomial kernels, the curve is linear and it is consistent with the number of genes evaluated. Therefore, it is concluded that the model requires more iterations to converge when it has less information, explaining the high peaks in total execution time at the beginning of graphs in Fig. 2.

Regarding the C-Index values obtained during the training phase, all kernels suffer a degradation in performance as the number of features increases (Fig. 4). This may be because the dataset used is highly censored and, therefore, unbalanced: it consists of 86% of the samples without event occurrence (931 samples) and only 14% with event definition (151 samples). Furthermore, the fact that the data is not linearly separable explains the poor performance of the linear kernel.

Expanding the experiments by adjusting the C parameter of the Survival-SVM or changing the distribution function to norm-1, which is preferred with highdimensional data, could yield better results. Different techniques could also be applied to address the imbalance problem, as proposed in [14].

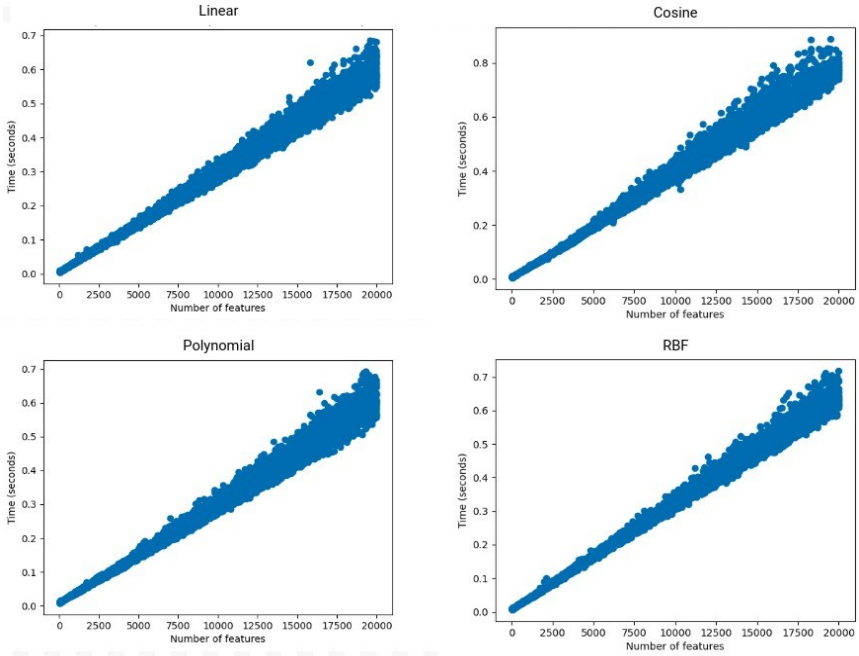


Fig. 1. Average test times during cross-validation for the Linear, Cosine, Polynomial, and RBF kernels.

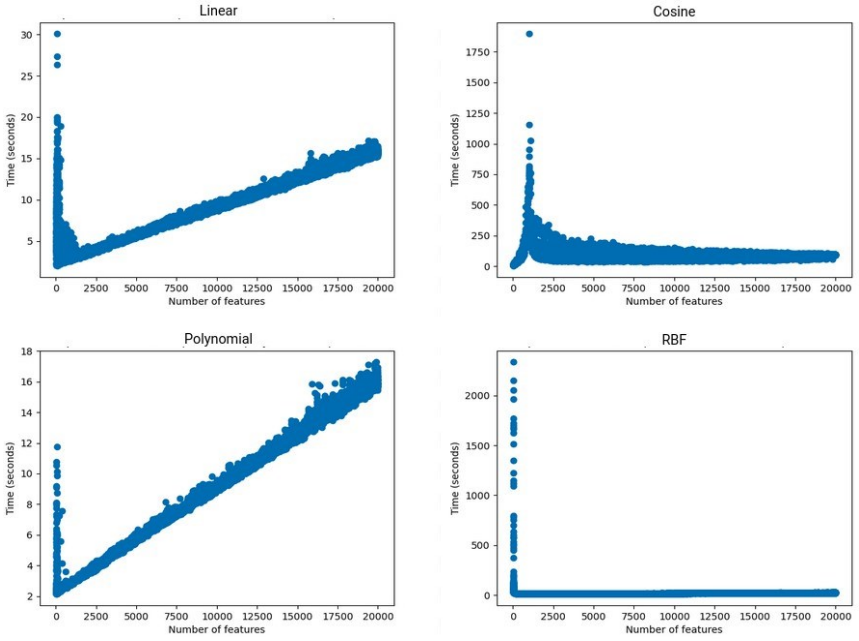


Fig. 2. Total execution time of the 10-fold CV process for Linear, Cosine, Polynomial, and RBF kernels.

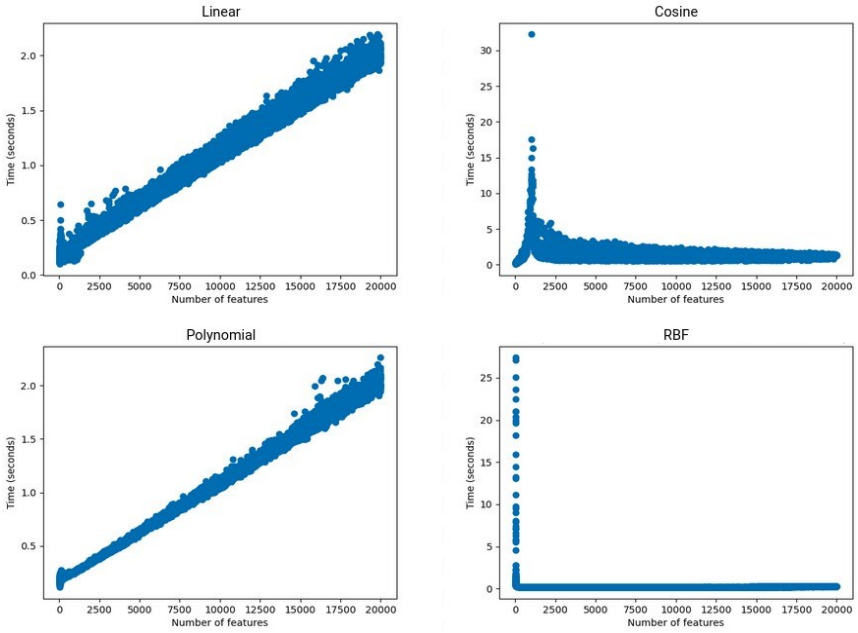


Fig. 3. Times per iteration for Linear, Cosine, Polynomial, and RBF kernels.

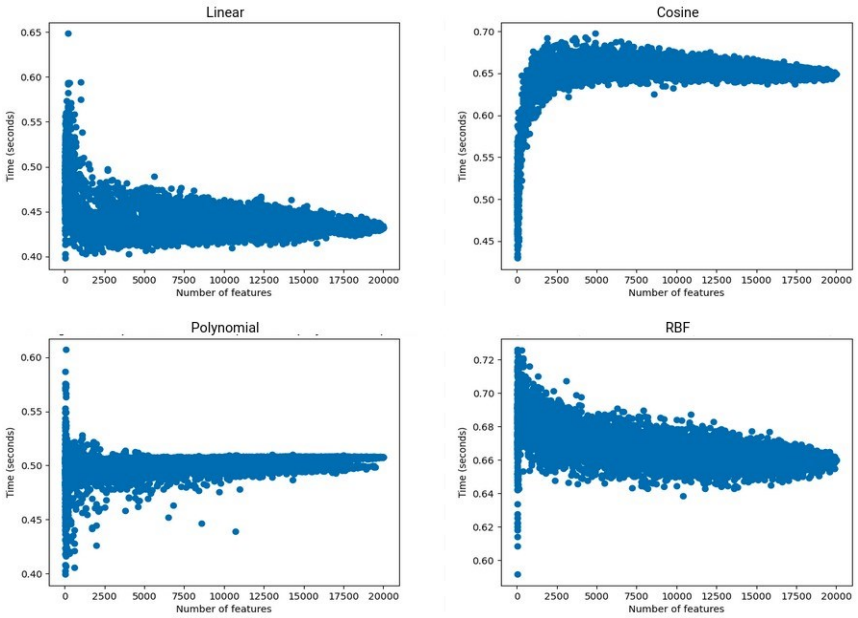


Fig. 4. C-Index obtained during the training phase of the Survival-SVM for Linear, Cosine, Polynomial, and RBF kernels.

4. Conclusion

The execution times and the C-Index obtained by the Survival-SVM model for different numbers of features in a gene dataset have been evaluated. The experiments were carried out with different kernels and optimizers, allowing a direct comparison between different configurations available.

Concluding, of the two available optimizers, RBTree generated a significant degradation in execution times during the training phase of the model in all cases. Regarding kernel configuration, worse time metrics were observed when the dataset had less than 100 features. Even so, iteration times remained directly correlated to dataset size for the Linear and Cosine kernels, making it clear that the high time required with few features is due to the number of iterations the model needs to reach convergence. It was not the case with the Polynomial and RBF kernels, whose times per iteration were also erratic for measurements with few training features.

The behavior of the model during the training stage was analyzed, and it was observed that the unbalanced nature of the data (a large amount of censored data) and the impossibility of separating them linearly resulted in worse metrics as the size of the dataset increased.

The results obtained show that, for subsets with few genes, the linear and polynomial kernels have a shorter execution time and an acceptable C-index. On the other hand, if the number of genes in the subset exceeds 2000, then the Cosine and RBF kernels obtain a better balance between execution time and prognostic power. Both tests show that it will be trivial to obtain models that predict the execution time of a model with N features, which will allow to improve task distribution in a distributed environment. Although the results presented in this work are indeed obtained from the analysis of a specific database, in the future it may include replicating the experiment in more databases to determine if there is variability.

The experiments were carried out on an Apache Spark cluster to reduce the time required to complete them, in addition to establishing a base configuration that will allow carrying these experiments to other models and continue advancing in the development of algorithms that will be used in production on the Multiomix platform for the application of FS techniques with survival data. Measuring these times will allow carrying out a comprehensive analysis, and establish a load balance algorithm within the computer cluster to obtain an optimal algorithm distribution and reduce execution times.

References

1. Khanesar, M.A., Teshnehlab, M., Shoorehdeli, M.A. A novel binary particle swarm optimization. 2007 Mediterranean Conference on Control & Automation, pp. 1-6, doi: 10.1109/MED.2007.4433821 (2007)
2. Pashaei, E., Aydin, N. Binary black hole algorithm for feature selection and classification on biological data. Applied Soft Computing, 56, 94-106 (2017)
3. Camele, G., Menazzi, S., Chanfreau, H., Marraco, A., Hasperué, W., Butti, M. D., Abba, M.C. Multiomix: a cloud-based platform to infer cancer genomic and epigenomic events associated with gene expression modulation. Bioinformatics, 38(3), 866-868 (2022)

4. Ishwaran, H., Kogalur, U. B., Blackstone, E. H., Lauer, M. S. Random survival forests. The annals of applied statistics, 2(3), 841-860 (2008)
5. Camele, G., Hasperué, W., Ronchetti, F., Quiroga, F. M. Statistical analysis of the performance of four Apache Spark ML algorithms. Journal of Computer Science and Technology, 22(2), e14-e14 (2022)
6. Moncada-Torres, A., van Maaren, M. C., Hendriks, M. P., Siesling, S., Geleijnse, G. Explainable machine learning can outperform Cox regression predictions and provide insights in breast cancer survival. Scientific Reports, 11(1), 1-13 (2021)
7. Pölsterl, S., Navab, N., Katouzian, A. An efficient training algorithm for kernel survival support vector machines. arXiv preprint arXiv:1611.07054 (2016)
8. Pölsterl, S., Navab, N., Katouzian, A., Fast Training of Support Vector Machines for Survival Analysis, Machine Learning and Knowledge Discovery in Databases: European
9. Conference, ECML PKDD 2015, Porto, Portugal, Lecture Notes in Computer Science, vol. 9285, pp. 243-259 (2015)
10. Pölsterl, S., Navab, N., Katouzian, A., An Efficient Training Algorithm for Kernel Survival Support Vector Machines 4th Workshop on Machine Learning in Life Sciences, 23 September 2016, Riva del Garda, Italy (2016)
11. Botev, Z., Ridder, A. Variance reduction. Wiley statsRef: Statistics reference online, 1-6 (2017)
12. Harrell, F. E., Lee, K. L., Mark, D. B. Multivariable prognostic models: issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. Statistics in Medicine 15, 361–387 (1996)
13. Harrell, F. E., Califf, R. M., Pryor, D. B., Lee, K. L., Rosati, R. A. Evaluating the yield of medical tests. The Journal of the American Medical Association 247, 2543–2546. (1982)
14. Liu, J., Lichtenberg, T., Hoadley, K. A., Poisson, L. M., Lazar, A. J., Cherniack, A. D., Kovatich, A. J., Benz, C. C., Levine, D. A., Lee, A. V., Omberg, L., Wolf, D. M., Shriver, C. D., Thorsson, V., Cancer Genome Atlas Research Network, Hu, H. An Integrated TCGA Pan-Cancer Clinical Data Resource to Drive High-Quality Survival Outcome Analytics. Cell, 173(2), 400–416.e11. https://doi.org/10.1016/j.cell.2018.02.052 (2018)
15. Batuwita, R., Palade, V. Class imbalance learning methods for support vector machines. Imbalanced learning: Foundations, algorithms, and applications, 83-99 (2013)

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Replicación sincrónica y asincrónica de Bases de Datos en Modalidad "Maestro-Esclavo", caso de estudio: Firebird SQL Server

Guillermo Ruben Cherencio[1], Mario Gerardo Perello[2], Juan Carlos Romero[3]

[1] Universidad Nacional de Lujan, Depto, de Ciencias Básicas,
gcherencio@unlu.edu.ar

[2] Universidad Nacional de Lujan, Depto. de Ciencias Básicas,
mperello04@yahoo.com.ar

[3] Universidad Nacional de Lujan, Depto. de Ciencias Básicas,
juancarlosjromer@gmail.com

Abstract. Los Sistemas de Gestión de Bases de Datos (SGBD) más evolucionados permiten la programación de desencadenadores o disparadores, procedimientos almacenados y funciones externas. En nuestro caso práctico: Firebird/Interbase® (FB) incorpora el lenguaje de programación PL/SQL, el cual permite la programación de disparadores y procedimientos almacenados. FB también permite declarar funciones externas definidas por el usuario (UDF), que pueden escribirse en distintos lenguajes de programación. Una vez declarada una función UDF, puede utilizarse como una función propia de FB y puede ser invocada desde cualquier disparador escrito en PL/SQL. Utilizando estas características, se desarrolló el proyecto de investigación sobre replicación de bases de datos no distribuidas denominado "Replicación maestro-esclavo en bases de datos relacionales no distribuidas"[1], en este proyecto se implementaron dos formas de replicación homogénea de tipo maestro – esclavo para FB: una basada en disparadores (replicación sincrónica, RS)[2] y otra basada también en disparadores pero usando una función UDF que actúa como cliente de un servidor de replicación (replicación asincrónica, RA)[3]. Ambas replicaciones cuentan con una granularidad de replicación a nivel de tabla y han requerido del desarrollo e implementación de un software que genere todo el código PL/SQL necesario dentro de la base de datos maestro y en cada una de las bases de datos esclavas, de forma tal, de facilitar y automatizar su implementación, evitando errores de codificación en PL/SQL. El presente trabajo se enmarca dentro de la etapa final del proyecto; se pretende realizar una prueba de uso intensivo de cada replicación y obtener los resultados de las mismas. Se abren futuras líneas de investigación sobre nuevas funcionalidades a incorporar a la aplicación propuesta para facilitar la implementación de bases de datos distribuidas en FB. Ambas replicaciones se desarrollaron bajo licencia LGPL.

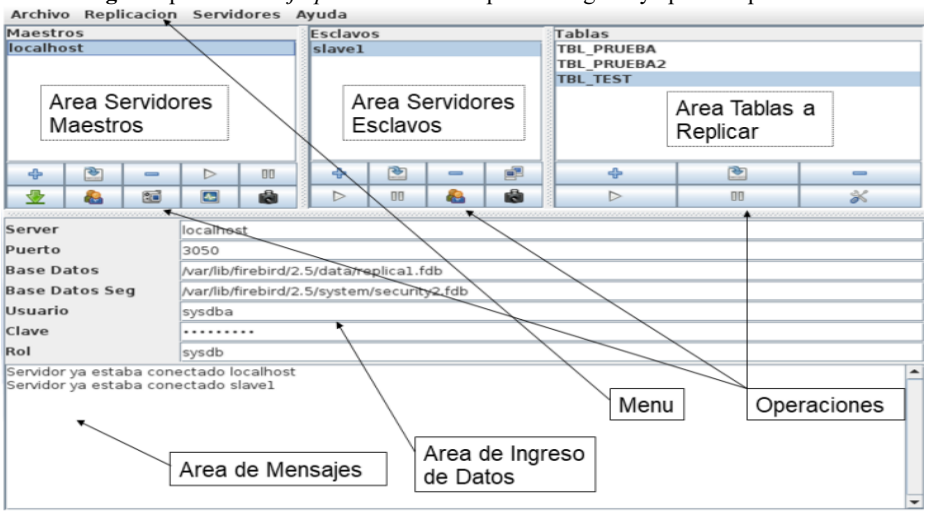
Keywords: Firebird Interbase Replicación Sincrónica Asincrónica SQL PL/SQL.

1. Introducción

En el marco del proyecto de investigación "Replicación maestro-esclavo en bases de datos relacionales no distribuidas" [1], se desarrollaron dos formas de replicación para FB: una replicación sincrónica basada en disparadores (RS) [2] y una replicación asincrónica también basada en disparadores (RA) [3]. Replicar una tabla con una

granularidad a nivel de tupla, bajo estos enfoques, implica implementar un disparador para cada evento de inserción, actualización y borrado de tuplas. En un sistema con 100 tablas a replicar, implicaría la implementación manual de unos 300 disparadores; cualquier modificación en la estructura de una tabla replicada, implicaría la modificación los disparadores asociados a la misma y la estructura de la tabla en las bases de datos esclavas que la repliquen. Por lo tanto, se desarrolló una aplicación con interfaz gráfica para que el usuario pueda configurar allí la replicación maestro – esclavo que pretende implementar, permite generar y aplicar todo el código PL/SQL necesario en servidor maestro y en cada uno de los servidores esclavos; esta aplicación la llamamos *fbjreplicator* y funciona para RS y RA:

Fig. 1. Aplicación *Fbjreplicator* utilizada para configurar y aplicar replicación.



La aplicación *fbjreplicator* solo se requiere para configurar, gestionar o instalar la replicación RS o RA; una vez instalada la replicación, la misma funciona sin necesidad de esta aplicación.

1.1 Replicación Sincrónica (RS)

A partir de la versión 2.5 FB incorpora el algoritmo 2PC[4] en la ejecución de comandos de tipo EXECUTE STATEMENT[5] haciendo posible la ejecución de comandos SQL dinámicos en forma remota, indicando los datos para gestionar la conexión, es posible ejecutar el comando SQL en otro servidor FB y luego verificar el resultado de la transacción distribuida. Utilizando *fbjreplicator* y estos recursos que nos provee FB se implementó esta forma de replicación maestro – esclavo basada en disparadores, totalmente desarrollada usando código PL/SQL.

La replicación cuenta con información de *metadata* en servidor maestro, en donde se indican los servidores esclavos a replicar, datos de conexión, tablas a replicar, etc; además se cuenta con un log para registrar el éxito o fracaso en las transacciones distribuidas que se llevaron adelante. Esta información es muy importante para luego

poder hacer *redo* en un servidor esclavo que estuvo fuera de servicio por algún período de tiempo y volver a ponerlo en línea. Cuando se agrega, borra o cambia una tupla en una tabla replicada, se ejecutan los disparadores correspondientes y éstos generan el código SQL a enviar a cada uno de los servidores esclavos utilizando el comando EXECUTE STATEMENT. Los disparadores ejecutan en forma remota el procedimiento P_APPLY en cada servidor esclavo, el cual recibe dos argumentos: el número de transacción y el código SQL a ejecutar. P_APPLY ejecuta el código SQL y guarda este número de transacción, el cual nos permite saber cuál es la última transacción que ejecutó el servidor esclavo:

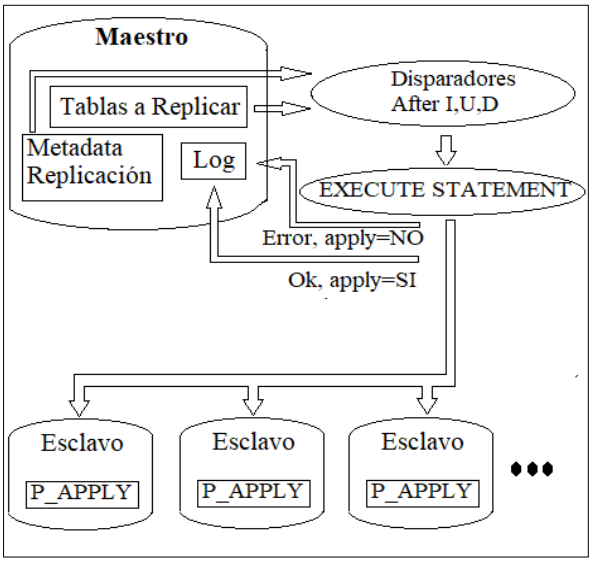


Fig. 2. Replicación maestro – esclavo sincrónica.

1.2 1.2 Replicación Asincrónica (RA)

Esta forma de replicación utiliza el diseño anterior, pero en vez de utilizar EXECUTE STATEMENT invoca una función UDF que envía los datos de replicación a un proceso servidor socket local de tipo AF_UNIX [6]. Las funciones definidas por el usuario (UDF) en FB se implementan como funciones externas escritas en algún lenguaje de programación (habitualmente C[7]), pueden utilizarse en distintos contextos y generalmente están limitadas a operaciones simples: manejo de cadenas de caracteres, cálculos, etc.[8]. Este trabajo ha permitido demostrar que es posible desarrollar una UDF que actúe como cliente de otro proceso servidor, a esta función la llamamos F_REPLICA. Esta función es cliente del proceso servidor FBSERVER el cual se implementó como un servidor socket local de tipo AF_UNIX, escrito en lenguaje C. FBSERVER recibe los datos de la replicación y ejecuta P_APPLY en cada uno de los servidores esclavos, utiliza la librería libfb[9] para la comunicación con FB:

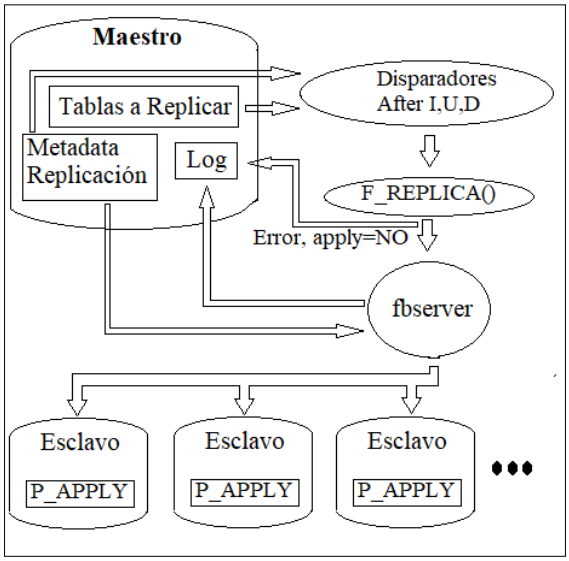


Fig. 3. Replicación maestro – esclavo asincrónica.

2. Recursos del trabajo y método utilizado

El presente trabajo pretende procesar 1.000.000 de transacciones de inserción, actualización y borrado de tuplas en tabla replicada utilizando RS y RA, de forma tal, de ver el funcionamiento general de estas formas de replicación, con un volumen de datos muy superior a las pruebas realizadas durante la etapa de desarrollo, que permita observar el rendimiento y depurar posibles errores de diseño o implementación:

1. Configurar y ejecutar replicación RS
2. Realizar ajustes y cambios necesarios para completar prueba
3. Registrar resultados
4. Configurar y ejecutar replicación RA
5. Realizar ajustes y cambios necesarios para completar prueba
6. Registrar resultados
7. Analizar resultados obtenidos

No se estableció ninguna prueba con respecto a consultas, debido a que en estas formas de replicación, todos los nodos pueden consultarse en forma local y todo puede reducirse a simple consulta cliente – servidor. En este modelo, lo relevante es la replicación de las operaciones de inserción, actualización y borrado.

2.1 Ejecución RS y RA

Para ambas replicaciones se optó por utilizar la misma máquina virtual, sin cambiar su configuración ni ubicación. Se utilizaron los recursos del Centro de Investigación,

Docencia y Extensión en Tecnologías de la Información y la Comunicación, de la Universidad Nacional de Luján (CIDETIC). Para evitar demoras en la replicación que pudiesen relacionarse con la red de área local, se optó por utilizar el mismo servidor FB local con bases de datos esclavas alojadas en la misma máquina virtual. Las pruebas se ejecutaron con el mismo usuario local, en un entorno Linux®. También se verificó que tanto la máquina virtual como el servidor físico en el cual se ejecuta la misma, no estuviese con una carga adicional de trabajo.

La tabla a replicar tiene una clave primaria simple y columnas con tipos de datos distintos, todos los soportados por las replicaciones RS y RA, de forma tal, que también permite probar la generación de código PL/SQL.

Existen 3 tipos de transacciones: de inserción, de borrado y de actualización. Todas afectan a una única tupla de la tabla a replicar.

Se implementó un único programa (*fbjrtest*) para lanzar las transacciones y medir el tiempo de ejecución del mismo, utilizando el comando *time* de Linux®. El programa permite crear las transacciones distribuidas de inserción, borrado y actualización.

3. Resultados

El comando *time* de Linux® emite tres resultados en la ejecución de un proceso: *real* se refiere al tiempo total de proceso, incluyendo operaciones de entrada – salida (I/O); *user* se refiere al tiempo que el proceso se ejecutó en modo usuario[10], sin incluir el tiempo bloqueado en operaciones de entrada – salida; *sys* se refiere al tiempo que el proceso se ejecutó en modo super-usuario (también llamado modo *kernel* o privilegiado)[10], sin incluir el tiempo bloqueado en operaciones de entrada – salida.

En la Tabla 1, la columna Total es igual a *real*; la columna Ejecución es igual a *user* más *sys*; la columna I/O es igual a Total menos Ejecución. La ejecución de la replicación RS pudo realizarse sin requerir mayores cambios y se obtuvieron los siguientes resultados para 1.000.000 de transacciones de cada tipo:

Tabla 1. Tiempos de ejecución (en segundos) de RS

Tipo de Transacción	Total	Ejecución	I/O
Inserción	159.638,236	228,772	159.409,464
Actualización	150.460,605	239,396	150.221,209
Borrado	143.126,56	205,456	142.921,104

En cuanto a RA, las dificultades para llevar adelante la prueba se centraron en FBSERVER, proceso servidor socket local, escrito en lenguaje C[7], el diseño original del mismo había previsto la utilización de sockets de tipo AF_UNIX[6] y basado en hilos (*threads*), utilizando la librería pthread (*POSIX Threads*)[11]. Las pruebas no lograban superar las 10.000 transacciones, se observaba un bajo rendimiento y un excesivo consumo de memoria que terminaba en errores de ejecución asociados a la librería pthread. Existe un costo de memoria asociado a la creación de cada hilo que crea el servidor para atender a cada transacción que arriba a FBSERVER. Este costo es razonable para procesar un número menor a 10.000 transacciones (teniendo en cuenta

los recursos disponibles de la máquina virtual), pero cuando se desea procesar 1.000.000 de transacciones que se ejecutan en un lapso de tiempo muy corto, el tiempo de proceso de cada transacción no permite dar tiempo a la liberación de recursos pthread por la finalización de los hilos de las transacciones ya procesadas. La solución fue establecer un parámetro en FBSERVER para crear un *pool* de hilos configurable acorde con los recursos disponibles y no crear más hilos una vez que se ha completado todo el *pool*; en tal caso, se debe esperar hasta la finalización de hilos permitan el reuso de su ubicación en el *pool*. Habiendo resuelto el problema de pthread, se observa que FBSERVER no logra atender con la suficiente velocidad la carga de trabajo que implican 1.000.000 de transacciones, se optó por cambiar el servidor socket de tipo AF_UNIX[6] por uno de tipo AF_INET[6], esto permitió bajar considerablemente las operaciones de entrada – salida de FBSERVER, mejorando el rendimiento y por lo tanto, reducir el tiempo de espera de las transacciones que arriban cuando el *pool* de hilos está al 100% de ocupación.

Se continuó con la misma forma de medición que en RS, utilizando el comando *time* de Linux® en la ejecución del programa *fbjrtest*, con el objeto de medir el rendimiento desde el punto de vista del usuario, sin considerar el procesamiento que podría estar realizando FBSERVER luego de concluir *fbjrtest*.

La ejecución de la replicación RA pudo realizarse y se obtuvieron los siguientes resultados para 1.000.000 de transacciones de cada tipo:

Tabla 2. Tiempos de ejecución (en segundos) de RA

Tipo de Transacción	Total	Ejecución	I/O
Inserción	34.426,989	279,852	34.147,137
Actualización	25.750,479	286,86	25.463,619
Borrado	22.122,218	273,368	21.848,85

4. Discusión

El presente trabajo no pretende realizar una comparativa entre ambos sistemas de replicación sino observar su rendimiento, con el objetivo de que esto pueda ser relevante en la implementación de sistemas productivos utilizando alguna de las formas de replicación aquí expuestas.

Se observa tanto en RS (99.85%) como en RA (99,18%) que el tiempo de proceso corresponde mayoritariamente a operaciones de entrada-salida. Los tiempos de entrada – salida en RS son exagerados con respecto a RA, sin embargo, los tiempos de proceso en ejecución son muy similares entre RS y RA; siendo en RA siempre superiores a RS.

En las pruebas realizadas, en RA, luego de finalizado *fbjrtest*, se observa que en pocos segundos (no se ha podido mensurar con precisión) FBSERVER termina de procesar su pool; aunque sumemos este tiempo a los tiempos de RA, continua siendo la opción más rápida. FBSERVER y F_REPLICA() graban líneas de mensaje en archivos de texto para su posterior análisis, que podrían eliminarse o reducirse, mejorando aún más los tiempos de RA.

RA es –en promedio- 6 veces más rápida que RS y esta diferencia de velocidad se debe a las operaciones de entrada – salida de RS.

En ambas replicaciones, el orden de las operaciones, de las más lenta a las más rápida, han sido: inserción, actualización y borrado. Sin embargo, no se mantiene la misma proporcionalidad en cuanto a los tiempos de duración de las operaciones, por ejemplo, en RS una inserción tarda un 6% más que una actualización, mientras que en RA una inserción tarda un 33% más que una actualización; en RS una actualización tarda un 5% más que un borrado, mientras que en RA una actualización tarda un 16% más que un borrado.

5. Conclusión

Ambos tipos de replicación son factibles de implementar en producción. Es relevante tener en cuenta los requerimientos del usuario en cuanto al tipo de replicación requerida, el número de transacciones promedio a procesar por unidad de tiempo y la cantidad de servidores esclavos. Es crítico acelerar las operaciones de entrada – salida. RA es 6 veces más rápida que RS desde el punto de vista del usuario que envía las transacciones a procesar.

RS hace un uso intensivo de operaciones de entrada – salida, las posibles razones pueden ser:

- uso de cursor en disparadores para acceder a la *metadata* de replicación; si estas tablas no se encuentran en memoria, su uso intensivo implicaría muchas operaciones de entrada - salida.
- uso de la instrucción EXECUTE STATEMENT para enviar comandos a servidores remotos; la ejecución de SQL dinámico implica la interpretación del mismo, su compilación y ejecución, en el contexto de un disparador.
- implementación del protocolo 2PC en Firebird/Interbase®, los disparadores no permiten ejecución en paralelo, por lo tanto, para cada servidor esclavo, se construye y se envía el código SQL dinámico a procesar de forma secuencial, esto no permite aprovechar el procesamiento distribuido que podrían hacer los servidores esclavos y por cada servidor esclavo que se agregue a la replicación, se penalizarán a todas las transacciones distribuidas con mayor tiempo de procesamiento. No se cuenta con mayor información en cuanto a la implementación de 2PC, pero debe llevar la cuenta de la aplicación satisfactoria (o no) de cada transacción en cada servidor esclavo.

Se ha demostrado que es posible desarrollar una UDF que actúe como cliente de otro proceso servidor y soporte una alta carga de trabajo; esto alienta el desarrollo nuevos trabajos que permitan desacoplar a los disparadores sincrónicos, de forma tal, que los servidores esclavos puedan procesar las transacciones distribuidas en paralelo.

Si se requiere una replicación de alto rendimiento, con la arquitectura actual de Firebird/Interbase®, se debe recomendar la implementación de RA o variantes de la misma, que permitan una ejecución rápida de los disparadores, enviando la información básica de replicación a un servidor que lo procese en paralelo, utilizando una UDF.

6. Agradecimientos

Al Mgter. Jorge A. Peri, por su orientación y dirección; al Centro de Investigación, Docencia y Extensión en Tecnologías de la Información y la Comunicación, de la Universidad Nacional de Luján (CIDETIC) por facilitar los medios y recursos necesarios para la realización de proyecto.

References

1. Peri, Jorge A.; Cherencio, Guillermo R.; Perello, Mario G.; Romero, Juan C.; Proyecto de investigación "Replicación Maestro-Esclavo en Bases de Datos Relacionales No Distribuidas", Universidad Nacional de Luján, Argentina, Departamento de Ciencias Básicas, Resolución 418/17, https://resoluciones.unlu.edu.ar/documento.frame.php?cod=87611.
2. herencio, Guillermo R.; Perello, Mario G.; Romero, Juan C.; "Replicación de Bases de Datos en Modalidad maestro-esclavo, caso de estudio: Firebird SQL Server", CACIC 2018, Tandil, http://cacic2018.exa.unicen.edu.ar/pdf/LibroDeActasCACIC2018.pdf , Editorial: Red de Universidades Nacionales con carreras en Informática (RedUNCI), ISBN 978-950-658-472-6
3. Cherencio, Guillermo R.; Perello, Mario G.; Romero, Juan C.; "Replicación de tipo Maestro - Esclavo Asincrónica en Firebird/Interbase SQL Server", 7mo Congreso Nacional de Ingeniería Informática / Sistemas de Información, CONAIISI, 2019, Universidad Nacional de La Matanza, UNLaM, https://conaiisi2019.unlam.edu.ar/ , ISSN: 2347-0372, ISBN 978-987-4417-73-2
4. Rahimi, Saeed K.; Haug, Frank S, "Distributed Database Management Systems. A practical approach", John Wiley & Sons Inc. Publication, IEEE Computer Society, 2010, ISBN: 978-0-470-40745-5, Pag. 329-345.
5. Vinkenoog, Paul et al., "Firebird 2.5 Language Reference Update", Oct 8 2011, disponible en http://www.firebirdsql.org/file/documentation/reference_manuals/reference_material/html/langrefupd25.html
6. Stevens, Richard W., "Advanced Programming in the Unix Environment", Pearson Education, 1993, ISBN 0-201-56317-7, Pag. 496-514
7. Kernighan Brian W., Ritchie Dennis M, "El Lenguaje de Programación C", Segunda Edición, Pearson Educación, ISBN 968-880-205-0
8. Gregory Deatz, "User-defined Functions for InterBase", Borland Developers' Conference, San Diego, July 2000, disponible en: https://firebirdsql.org/en/writing-udfs-for-interbase/
9. Cherencio, Guillermo R, "Envolviendo la API de Firebird/Interbase SQL Server", 1er Congreso Nacional de Ingeniería Informática / Sistemas de Información, CONAIISI, 2013, Editorial: Red de Carreras de Ingeniería Informática / Sistemas de Información, Consejo Federal de Decanos de Ingeniería, Cordoba, ISSN: 2347-0372, Disponible en http://conaiisi.frc.utn.edu.ar/PDFsParaPublicar/1/schedConfs/3/9-447-1-DR.pdf
10. Stallings, William, "Sistemas Operativos, Aspectos internos y principios de diseño.", Pearson, Prentice Hall, Madrid, 5ta. Edición, 2005, ISBN 84-205-4462-0, Pag. 135-137.
11. Butenhof, David R., "Programming with POSIX Threads", Addison-Wesley, 1997, ISBN 978-0-201-63392-4.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

On the Assessment of Personality Traits by Using Text Mining Techniques

Luis Montenegro[1], Maximiliano Sapino[1,2,3], Edgardo Ferretti[1,2], and Leticia Cagnina[1,2,3]

[1] Universidad Nacional de San Luis (UNSL), San Luis - Argentina
[2] Laboratorio de Investigación y Desarrollo en Inteligencia Computacional, UNSL
[3] Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET) - Argentina
{ferretti,lcagnina}@unsl.edu.ar

Abstract. This paper reports a complete experience of the Knowledge Discovery in Databases process to solve a personality trait assessment problem using text mining techniques. Given that this work is part of an interdisciplinary study between researchers from the fields of Computer Science and Psychology, in this first approach, four simple predictive algorithms were evaluated; namely: Multinomial Naive Bayes, Logistic Regression, Support Vector Machines and Decision Trees. Moreover, given the nature of the problem faced, where one person may present more than one personality trait, but not necessary all of them, it was modeled by three different classification tasks: viz. binary, multiclass and multilabel. Besides, data augmentation was used as a useful technique to improve the performance of all the classification approaches evaluated. Particularly, binary classification was the approach which took more advantage of using this technique by improving its performance on average by 13% compared to the original dataset. For three out of the five personality traits studied, it achieves weighted-F_1 scores above 0.75 and in particular the highest score of 0.88 was achieved for the *Responsibility* trait.

Keywords: Personality Traits · Big Five Factors · Knowledge Discovery in Databases · Text Mining · Data Augmentation.

1. Introduction

Personality psychology focuses on the study of individual differences and psychological traits. This perspective holds that there are certain patterns of personality that determine certain behavioral styles [3]. There is also evidence that spoken and written language tends to be stable over time, and it holds words that people use to describe themselves and others, so it is a useful way to access people's personality [5]. In this context, *The Big Five Factors* (BFF) model is based on the premise that the language that people daily use describes their personality. Thus, personality can be understood as the expression of five main traits: *Neuroticism*, *Extraversion*, *Openness to experience*, *Agreeableness* and *Responsibility*.

Some studies have reported that language can be used to study and assess personality, for example language from social media or written texts [20]. For

instance, extraversion has been related to words referring to family, social activities and assertiveness. Agreeableness is identified with speeches that reflect people inclusion, words related to family, interpersonal concern, altruism and modesty. On the

other hand, responsibility is shown by using words related to achievements and work. Finally, openness to experience is not usually related to certain words of content, but rather with the use of particular articles and prepositions [23].

Moreover, there are studies (cf. [22]) that state that the natural language that people use on a daily basis not only reflects their personality, but also exposes their social situation and the interpersonal relationships they establish with others. Likewise, they also state that language can be used to assess people's physical and mental health. Other studies such as [4], suggest that *Text Mining* (TM) tools could be used even for early detection and prevention of suicide. This is due to the fact that the psychological interpretation assumes that the degree to which a person uses a category of words reflects how important that topic is for him/her, making explicit the importance of performing this kind of studies.

This context evidences the need to carry out interdisciplinary research, between Psychology and Computer Science (CS) [13,21]. In particular, since 2017, the *eRisk* international competitions have been annually organized [8–11,16–18] to try to automatically detect from written texts –by means of TM techniques– mental illnesses such as depression, anorexia and self-harm. And it should be noted that the BFF model is the theory generally adopted by research on CS that analyzes personality by applying data mining techniques [15,21].

Finally, it is worth mentioning that, in the specific field of psychological assessment, one of the main difficulties when working with the specific written material (the questionnaires) is the time and resources required to collect and analyze the data. In [2], the study concluded that 45% of the time is used exclusively in administrating the instruments, 36% in their evaluation and the remaining 19% in the interpretation of results. For these reasons, some therapists end up delegating these tasks to lesser experienced professionals. In this respect, the development of automatic systems based on TM methods, speeds up the time spent in interpreting the results and makes possible the measurement or evaluation of alternative constructs, different from the traditional questionnaires or surveys used.

Given the above context, our research hypothesis is based on the fact that, by means of TM techniques, static and dynamic characteristics can be extracted from free texts that will reflect the personality traits of people; and consequently to obtain a classifier that automatically estimates their personality traits. This approach differs from the traditional one which takes a lot of time, given that evaluation is manually done by professionals. Our research goal is to obtain equivalent evaluation methods in terms of quality prediction, with a high gain in the time consumed. With this aim, the following particular objectives must be accomplished: (1) to create a data corpus (consisting of free texts) of participants whose personality traits have already been psychologically evaluated by a professional psychologist; (2) to use appropriate document representations considering static and dynamics features, extracted by using TM techniques; (3) to obtain classifiers that automatically estimate the personality traits of the participants; and (4) to evaluate the predictive performance of the classifiers by comparing their results with the scores obtained by the participants in the BFF Inventory.

The rest of the article is organized as follows. Section 2 describes how our dataset was created. Then, in Section 3, we present the formal problem statement and, the

different classification approaches evaluated are mentioned. Also, the document models used to represent the articles are discussed. Section 4 reports on the experimental setting carried out and the obtained results. Finally, Section 5 offers the conclusions.

2. Dataset

Our survey is composed by 298 samples, i.e. 298 different people that completed the requested tasks. They are mainly students of the National University of San Luis but also 2.9% of the people are employed in the private industry and 3.3% work in the public sector. Considering their origin, 84% were born in the capital of San Luis and the remaining 16% were born in 19 different cities from San Luis province and other provinces from Argentina. All the people live in San Luis at present. Ages ranged from 17 to 54 years old, with an approximate mean of 22 years old. Regarding educational level, 96% of the participants have completed secondary education and only 4% have completed a universitary level. When gender is considered, 73% of the samples correspond to female and 27% to male individuals. Finally, it is worth mentioning that 93% of the people do not have children, while the remaining 7% do.

The aforementioned tasks that were requested to the people participating on the study consisted in completing the *Big Five Inventory* developed by [7] and adapted to the argentinian population by [3]. It consists of 44 items that participants rate on a *Likert*[1] scale depending on how strongly they identify with the statement. In addition, participants were asked to write three texts based on how they would describe themselves (how they generally feel, how they behave, etc.), a report of how they spent their last week (what they did, who they were with, and all they were able to tell) and how they imagine themselves in the next five years (what they expect to be doing, who would they see themselves with, etc.).

Then, by using the IBM SPSS[2] software (version 23), the raw scores are converted into percentiles, where a percentile of 75 or more indicates presence of the trait and is taken as a positive instance, and a percentile of 25 or less indicates absence of the trait and is taken as negative instance. Instances whose values fall in the range between 25 and 75 are discarded since they may introduce noise into the classifier training process since they cannot be considered –as we refer in colloquial language– as "strong" positives or negatives (or representative samples). All in all, the final sets for each personality trait have the following proportions:

Neuroticism (256 total instances):	135 (No) - 121 (Yes)
Extraversion (250 total instances):	170 (No) - 80 (Yes)
Openness to experience (240 total instances):	98 (No) - 142 (Yes)
Agreeablenes (224 total instances):	71 (No) - 153 (Yes)
Responsibility (240 total instances):	186 (No) - 54 (Yes)

[1] Likert scale allows to rate a question or an statement with a five-points response, where score 5 corresponds to "Strongly agree", 4 corresponds to "Agree", 3 is equivalent to "None", 2 implies "Disagreement" and 1 means "Strongly disagreement".

[2] https://www.ibm.com/es-es/spss

3. Problem Statement and Classification Approaches

In the context of classifying personality traits, the problem can be considered in principle as a concept learning problem (usually named as binary classification); that is to say, a specific classifier is trained to learn a personality and to be able to indicate whether a particular text corresponds or not to that trait (concept). However, it is also possible to consider it as a multiclass single-label or multiclass multi-label classification problem, since a person might possess one or more personality traits simultaneously. We address these three classification approaches. To simplify referentation, we will use the term *multiclass* for the case where a single label is assigned to a document by a classifier trained with all the classes, while we will refer with the term *multilabel*, the latter case.

As mentioned above, we evaluated four simple predictive algorithms but with a great explanatory power; namely: Multinomial Naive Bayes [1], Logistic Regression [6], Support Vector Machines [12] and Decision Trees [14]. In addition, good performance is provided with lower computational costs than deep neural approaches. Due to space constraints, we do not provide further details of these four approaches and the interested reader is referred to the previous citations of these methods.

3.1 Document Models

Vectorization is used to convert a text into a vector of numeric features. As previously mentioned, static and dynamic characteristics were used in this work to be able to analyze different vector representations. With respect to the dynamic representation, Bag of Words (BoW) was the undelying representation model with Binary, TF-IDF and TF vectorization of the features' values. In order to bound the vectors' dimensionality, the 500 most frequent terms appearing in at least two documents were used, thus discarding more than 80% of the terms appearing in all the documents.

Regarding the static representation, the LIWC linguistic analyzer was used to compute the values of the 90 features of each text and thus obtain the numerical vector that constitutes the representation of each text in the dataset. In our case, the 2015 version has been used [19]. For each text, LIWC2015 reads one target word at a time. As each target word is processed, is looked up in the dictionary—which is made up of nearly 6,400 words, root words, and selected emoticons. If the target word matches a word in the dictionary, it will increases the scale (or scales) associated with the category of the word; for example, the word *cried* is part of five word categories: *sadness*, *negative emotion*, *general affect*, *verbs* and *past focus*. Therefore, if this word is found in the target text, each of these five categories will increase the scores on the subdictionary scale (which it is generated while texts are processed in the collection).

4. Experiments and Results

This section explains how the text classification experiments were carried out. For this purpose, data preparation is detailed, including data augmentation, preprocessing

and selection of relevant features for the models. Then, the training and validation processes of the models are described. Finally, we present the results obtained in the test stage performed. To properly contextualize the study, a brief description of the working environment used is also provided.

4.1 Experimental Setting

Google Colaboratory was used as working environment to carry out the experiments and the programming language used was Python in its stable version 3.8. In order to build the classification models, *sckit-learn* library tools were used. As above-mentioned in Sect. 2, our original dataset was composed by 298 samples, but 14 of them were removed since they were completed with random sentences to reach the minimun amounts of words each field required in the questionnaries.

For the remaining 284 texts of the original dataset, the following preprocessing techniques were applied: removal of punctuation marks, correction of spelling errors, removal of numbers and elimination of characters that were repeated at least three times. Additionally, the normalization of the texts was carried out to a simpler format such as the transformation of all texts to lowercase. Moreover, stop words (SW) removal was also performed.

It was decided to use 5-fold cross-validation, given that it is an appropriate method due to the limited availability of data and the need of obtaining more precise estimations of the performance of the classification models. As performance metric, we decided to use the F_1 measure since, in combination with cross-validation, it allows calculating the so-called weighted-F_1 measure. With this adjustment, an averaged F_1 score is obtained, weighted according to the distribution of classes (positive or negative labels) in the different folds of the cross-validation approach. This metric provides a more balanced assessment of the model's performance compared to a simple F_1 measure; especially when the dataset has unbalanced classes, as it is our case.

Data Augmentation As will be discussed in Sect. 4.2, the classification results for the original dataset were not as good as originally expected. Although several factors may influence the low performance of the classifiers, in our view, as aforesaid, data imbalance in the classes (people with and without the personality trait) could be one of the most important. To assess this issue, data augmentation techniques were applied, which consist of generating new training data from the original data.

Backtranslation is a technique which consists of translating a text in a source language (Spanish in our case) to a target language (viz. English) and then, translating back the texts into the source language, using a machine translation model. This back and forth process allows to generate new texts in the source language that are similar in structure and content to the original text, but that may vary in their wording and choice of words. On the other hand, GPT-2 (Generative Pre-trained Transformer 2), is a language model based on transformers that has been trained on large amounts of textual data to generate coherent and natural text.

In our study, backtranslation and GPT-2 were combined in the generation of new data. Even though both techniques can be used separately, since the text generated with backtranslation is quite similar to the original one and the texts generated with the GPT-2 model in Spanish are of low quality, it was decided to combine both techniques in the

following way: the texts in Spanish which are translated into English are used in turn as input for GPT-2 to generate new texts in English which are translated back into Spanish. Then, these new augmented dataset was preprocessed in the same way than the original one, and document models were built following the same procedure mentioned in Sect. 3.1. The addition of these new data is expected to improve the performance of the models in the personality trait classification task. Data augmentation was applied in order to increase the number of texts from the minority class. Below, it is shown the increase in texts that was carried out in each personality trait for the binary classification case:

Neuroticism (augmented in 25 instances):	135 (No) - 146 (Yes)
Extraversion (augmented in 86 instances:)	170 (No) - 166 (Yes)
Openness (augmented in 67 instances):	165 (No) - 142 (Yes)
Agreeablenes (augmented in 63 instances):	134 (No) - 153 (Yes)
Responsibility (augmented in 222 instances):	186 (No) - 276 (Yes)

For the multiclass and multilabel cases, the principle to generate new texts was the same one mentioned above, taking into account that if a text *T* was labeled as *[yes, no, no, yes, no]*, where each position correspond to a personality trait, the texts generated for that instance will have the same labels. Thus, the total number of new created instances per trait was 161. Table 1 shows the amount of instances in the original dataset and in the augmented one. As can be observed, in the original dataset for multiclass classification, the number of instances are fewer with respect to binary classification. This is due to the fact that for the multiclass classification approach there must be considered the instances that have "strong" labels (positive or negative) for all the personality traits. In Sect. 2, it was described how the instances whose percentile values for a particular trait were in the range between 25 and 75 were not considered, given that they might introduce noise in the learning process. In this way, all the instances included in the original dataset for the multiclass classification task, were those whose percentile values for all the traits were lower or equal than 25, or higher or equal than 75.

Table 1. Number of instances in the original and augmented datasets for the multiclass classification approaches.

Traits		**Agreeableness**	**Openness**	**Extraversion**	**Neuroticism**	**Responsibility**
Original	**Yes**	88	79	40	67	29
	No	44	53	92	65	103
Augmented	**Yes**	143	144	70	152	114
	No	154	153	227	145	183

4.2 Results

In Sect. 4.1, data augmentation was introduced and described as part of our experimental study with the aim of increasing the low performance of our original models. As can be observed in Table 2, the classification results for the original dataset do not exceed the 70% of weighted-F_1 for four out the five personality traits studied—

responsibility was the only trait were binary classification achieved a weighted-F_1 above 80%. The results reported in this table were obtained with different representation models. All of them were obtained with a particular weighting scheme (Binary, TF-IDF or TF) of BoW models since the static representation of LIWC perform worse than expected. Values in bold, represent the best performing scores achieved by a particular classification approach for each personality trait.

For instance, DTC achieves the best value (0.649) for *Agreeableness* in the binary classification approach and for the multiclass case. For the binary classification, this score was achieved with the binary vectorization of the features' values, while for the case of multiclass, when TF-IDF was used instead. *Openness* was the personality trait for which, in general, all the models performed worst. LR achieved the best value of 0.632 for the multilabel case, when TF weighting scheme was used. When *Extraversion* is considered, DTC achieved the best score of 0.692 in a binary classification fashion using TF as weighting scheme. DTC also achieved the best score of 0.802 in a binary classification for the *Responsibility* trait also with TF weighting scheme. Finally, MNB reported the best value of 0.647 for *Neuroticism* for the multiclass case with the binary weighting scheme.

When considering the results for the augmented dataset (Table 3), we still can see that *Openness* was the personality trait for which the worst results were obtained, but in general all the classifiers tend to perform better for this trait with respect to the original dataset, and the same conclusion can be drawn for the remaining traits. Now, for four out the five personality traits the 70% of weightedF_1 is a lower bound score. Indeed, *Agreeableness* and *Extraversion* scores round 75% and *Responsibility* reports the best performing value of 0.879. In this case DTC was not the best classifier as for the original dataset. LR reported the best scores for the multilabel approach for *Openness* trait with a binary weighting scheme, while for the case of *Neuroticism*, TF weighting scheme was the best one. MNB also reported the best values for two traits, viz. *Extraversion* and *Responsibility* but in the binary classification fashion; for the former trait TFIDF weighting scheme was used, while for the latter TF performed best. Finally, SVM achieves the best score for *Agreeableness* trait also in a binary classification approach with TF-IDF weighting scheme.

Table 2. Comparison of original dataset models - Weighted F_1 measure.

Classifier		**Agreeableness**	**Openness**	**Extraversion**	**Neuroticism**	**Responsibility**
Binary Class	**MNB**	0.606	0.591	0.641	0.601	0.709
	SVM	0.614	0.622	0.645	0.575	0.714
	DTC	**0.649**	0.623	**0.692**	0.620	**0.802**
	LR	0.606	0.618	0.638	0.543	0.709
Multi Class	**MNB**	0.643	0.568	0.667	**0.647**	0.687
	SVM	0.535	0.486	0.574	0.531	0.687
	DTC	**0.649**	0.573	0.591	0.559	0.681
	LR	0.643	0.541	0.669	0.615	0.689
Multi Label	**MNB**	0.644	0.568	0.665	0.644	0.695
	SVM	0.617	0.564	0.656	0.611	0.684
	DTC	0.518	0.564	0.609	0.567	0.680
	LR	0.487	**0.632**	0.588	0.598	0.736

Table 3. Comparison of augmented dataset models - Weighted F_1 measure.

Classifier		Agreeableness	Openness	Extraversion	Neuroticism	Responsibility
Binary Class	**MNB**	0.752	0.656	**0.749**	0.613	**0.879**
	SVM	**0.757**	0.666	0.743	0.643	0.874
	DTC	0.734	0.663	0.716	0.586	0.846
	LR	0.717	0.663	0.737	0.610	0.834
Multi Class	**MNB**	0.668	0.628	0.709	0.649	0.679
	SVM	0.704	0.639	0.707	0.700	0.775
	DTC	0.607	0.572	0.648	0.585	0.668
	LR	0.682	0.656	0.747	0.698	0.804
Multi Label	**MNB**	0.669	0.626	0.716	0.649	0.681
	SVM	0.707	0.659	0.722	0.693	0.788
	DTC	0.606	0.585	0.653	0.590	0.670
	LR	0.548	**0.677**	0.723	**0.704**	0.693

5. Conclusions

Tables 2 and 3 show that using data augmentation clearly improves the classifiers performance considering all the personality traits in at least a 10%. The binary approaches take more profit on the augmented dataset by increasing their performance on average by 13% when compared to the original dataset, while the multiclass and multi-label improve their performances on average by 10.5% and 8.45%, respectively. On average, considering all the personality traits, the binary classification approaches achieved weighted-F_1 scores above 0.7. In particular, for *Agreeableness* and *Extraversion* scores round 0.75 and for *Responsibility* the best performing value reported is 0.879.

DTC was the best classifier for the original dataset, since it reported three of the best values achieved for *Agreeableness*, *Extraversion* and *Responsibility* traits, but for this latter one, it was achieved a good classification performance above 0.8 while for the remaining traits, all the obtained values were below 0.7. For the augmented dataset, MNB and LR provided the best results for four of the five traits, while SVM reported the remaining best score for *Agreeableness* trait. It was clear that the static document model chosen did not perform as expected and BoW clearly outperformed it. With respect to the weighting schemes used, binary and TF seem to be more appropriate than TF-IDF, but more studies should be conducted to drawn proper conclusiones on which weighting scheme performs best in general.

Acknowledgments. This work was partially supported by CONICET and Universidad Nacional de San Luis (PROICO P-31816).

References

1. Anguiano-Hernández, E.: Naive bayes multinomial para clasificación de texto usando un esquema de pesado por clases. Tech. rep., INAOE (2009)
2. Camara, W.J., Nathan, J.S., Puente, A.E.: Psychological test usage: Implicationsin professional psychology. Professional Psychology: Research and Practice 31(2), 141–154 (2000)
3. Castro Solano, A.: Técnicas de evaluación psicológica para ámbitos militares. Ediciones Paidós Ibérica, Buenos Aires, Argentina, 1st edn. (2005)
4. Desmet, B., Hoste, V.: Emotion detection in suicide notes. Expert Systems with Applications: An International Journal 40(16), 6351–6358 (nov 2013)
5. Fast, L.A., Funder, D.C.: Personality as manifest in word use: correlations withself-report, acquaintance report, and behavior. Journal of personality and social psychology 94 (2), 334–346 (2008)
6. Hilbe, J.M.: Logistic Regression Models. Chapman & Hall (2017)
7. John, O.P.: Handbook of personality: Theory and research., chap. The "Big Five" factor taxonomy: Dimensions of personality in the natural language and in questionnaires, pp. 66–100. The Guilford Press (1990)
8. Losada, D.E., Crestani, F., Parapar, J.: CLEF 2017 erisk overview: Early risk prediction on the internet: Experimental foundations. In: Cappellato, L., Ferro, N., Goeuriot, L., Mandl, T. (eds.) Working Notes of CLEF 2017 - Conference and Labs of the Evaluation Forum. CEUR Workshop Proceedings (2017)
9. Losada, D.E., Crestani, F., Parapar, J.: Overview of erisk: Early risk prediction on the internet. In: Bellot, P., Trabelsi, C., Mothe, J., Murtagh, F., Nie, J., Soulier, L., San Juan, E., Cappellato, L., Ferro, N. (eds.) Experimental IR Meets Multilinguality, Multimodality, and Interaction- 9th International Conference of the CLEF Association (2018)
10. Losada, D.E., Crestani, F., Parapar, J.: Overview of erisk 2019 early risk prediction on the internet. In: Crestani, F., Braschler, M., Savoy, J., Rauber, A., Müller, H., Losada, D.E., Bürki, G.H., Cappellato, L., Ferro, N. (eds.) Experimental IR Meets Multilinguality, Multimodality, and Interaction - 10th International Conference of the CLEF Association (2019)
11. Losada, D.E., Crestani, F., Parapar, J.: erisk 2020: Self-harm and depression challenges. In: Jose, J.M., Yilmaz, E., Magalhães, J., Castells, P., Ferro, N., Silva, M.J., Martins, F. (eds.) Advances in Information Retrieval - 42nd European Conference on IR Research, ECIR 2020 (2020)
12. Ma, Y., Guo, G. (eds.): Support Vector Machines Applications, chap. Chapter 2: "Multi-Class Support Vector Machine". Springer-Verlag (2014)
13. Mariñelarena-Dondena, L., Ferretti, E., Maragoudakis, M., Sapino, M., Errecalde, M.: Predicting depression: a comparative study of machine learning approaches based on language usage. Panamerican Journal of Neuropsychology 11(3), 42–54 (2017)
14. Mitchell, T.M.: Machine learning. McGraw-Hill Series in Computer Science, McGraw-Hill (1997)
15. Moreno, J.D., Martínez-Huertas, J., Olmos, R., Botana, G., Botella, J.: Can personality traits be measured analyzing written language? a meta-analytic study on computational methods. Personality and Individual Differences 177 (2021)
16. Parapar, J., Martín-Rodilla, P., Losada, D.E., Crestani, F.: erisk 2021: Pathological gambling, self-harm and depression challenges. In: Hiemstra, D., Moens, M., Mothe, J., Perego, R., Potthast, M., Sebastiani, F. (eds.) Advances in Information Retrieval - 43rd European Conference on IR Research, ECIR 2021 (2021)
17. Parapar, J., Martín-Rodilla, P., Losada, D.E., Crestani, F.: Overview of erisk 2022: Early risk prediction on the internet (extended overview). In: Faggioli, G., Ferro, N., Hanbury, A.,

Potthast, M. (eds.) Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum (2022)
18. Parapar, J., Martín-Rodilla, P., Losada, D.E., Crestani, F.: ERISK 2023: EarlyRisk Prediction on the Internet - CLEF 2023 Workshop. https://erisk.irlab.org/ (Septiembre 2023), tesalónica, Grecia
19. Pennebaker, J.W., Boyd, R.L., Jordan, K., Blackburn, K.: The development andpsychometric properties of liwc2015. Tech. rep., The University of Texas at Austin (2015)
20. Pennebaker, J.W., Graybeal, A.: Patterns of natural language use: Disclosure, personality and social integration. Current directions in Psychological Science 10(3), 90–93 (2001)
21. Schwartz, H., Eichstaedt, J., Kern, M., Dziurzynski, L., Lucas, R., Agrawal, M.,Park, G., Lakshmikanth, S., Jha, S., Seligman, M., Ungar, L.: Characterizing geographic variation in well-being using tweets. In: Proceedings of the International AAAI Conference on Web and Social Media. vol. 7, pp. 583–591 (2021)
22. Tausczik, Y.R., Pennebaker, J.W.: The psychological meaning of words: Liwc andcomputerized text analysis methods. Journal of Language and Social Psychology 29(1), 24–54 (2010)
23. Yarkoni, T.: Personality in 100,000 words: A large-scale analysis of personality andword use among bloggers. Journal of Research in Personality 44(3), 363–373 (2010)

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Impacto de Factores Topológicos y de Desbalance en la Clasificación de Nodos con GCNs

Tatiana S. Parlanti[1], Carlos A. Catania[1], and Luis G. Moyano[2]

[1] Universidad Nacional de Cuyo, Facultad de Ingeniería, Laboratorio de Sistemas Inteligentes (LABSIN), Mendoza, Argentina

[2] División de Física Estadística e Interdisciplinaria, Centro Atómico Bariloche, Bariloche, Argentina

{tatiana.parlanti,harpo}@ingenieria.uncuyo.edu.ar

Resumen Las Redes Neuronales Convolucionales para Grafos (GCNs) han demostrado ser sumamente efectivas en la resolución de problemas relacionados con grafos, ya que no solo consideran las características individuales de los nodos, sino que también capturan la esencia topológica del grafo. No obstante, la escasez de conjuntos de datos públicos representa un obstáculo en la evaluación y comparación objetiva de estas redes en diversos contextos. En este artículo, se abordan las limitaciones inherentes de las GCNs, enfocándose específicamente en el impacto de los aspectos topológicos y el desbalance de clases en la tarea de clasificación de nodos. A través del algoritmo *Contextual Stochastic Block Model* (CSBM) se generan distintos grafos sintéticos que varían en características y topologías. De esta manera, se realiza una exploración exhaustiva de las capacidades de las GCNs en distintos escenarios. Los hallazgos iniciales remarcan la importancia fundamental de las características de los nodos en la clasificación y ponen de manifiesto los desafíos que surgen al tratar con escenarios de desbalance de clases.

Keywords: Redes Complejas, Redes Neuronales para Grafos, Grafos Sintéticos

1. Introducción

Un grafo $G = (V,E)$ es un conjunto de nodos o vértices V y un conjunto de enlaces o aristas E entre dichos nodos. El interés de estudiar estas estructuras es que muchos problemas se pueden interpretar a la luz de un grafo, describiendo apropiadamente quiénes son los nodos y qué representan las aristas que los unen. Así es que en los últimos años, las redes neuronales para grafos, conocidas como GNN por sus siglas en inglés (*Graph Neural Networks*), han tenido un marcado auge debido a su aplicación en numerosos problemas que involucran grafos, que van desde clasificación de proteínas según la estructura de la misma, hasta recomendación de contenido a usuarios en una plataforma, o predicción del tema del cual trata un paper teniendo en cuenta los trabajos que este cita y los que lo citan [3].

Uno de los primeros trabajos que exploró las redes neuronales para grafos fue el de Kipf y Welling [4], donde presentaron las redes convolucionales para grafos o GCN (*Graph Convolutional Networks*). Este trabajo tuvo gran impacto porque mostró que para un conjunto de nodos no etiquetados, se puede predecir su etiqueta usando no solamente las características de cada nodo, sino también -y allí radica lo novedoso- la información propia de las conexiones entre nodos. Es decir, el grafo subyacente que

describe el problema guarda información valiosa a la hora de entrenar y predecir usando una red neuronal.

Tomando como referencia el paper de Kipf y Welling [4], en este trabajo se tiene por objetivo explorar las GCNs a fin de determinar sus alcances y limitaciones. Para esto, se pondrá a prueba una GCN de dos capas, en diferentes conjuntos de grafos sintéticos, para clasificar los nodos de los mismos en un problema de clasificación binaria.

Lo que sigue está organizado de la siguiente forma: se comienza por una descripción teórica de las GNN, para seguir con una pequeña descripción del hardware y software utilizados, así como de los experimentos realizados. Luego se exponen los resultados y discusiones y finalmente las conclusiones y trabajo a futuro.

2. Breve descripción teórica de las GNNs

Antes de exponer los experimentos a realizar, se muestra a continuación una breve descripción teórica de las GNNs, que comienza por la notación a utilizar en todo el trabajo.

2.1 Notación

Dado un grafo $G = (V, E)$, se presentan a continuación varias definiciones relacionadas al concepto de grafo:

- Dos nodos $u, v \in V$ están conectados si $(u, v) \in E$.
- Dado un vértice $u \in V$, se define el entorno de u como $N(u) = \{v \in V \,|(u, v) \in E\}$.
- Un grafo se dice no dirigidos si $(u, v) \in E \Leftrightarrow (v, u) \in E$.
- Un grafo se dice dirigido si $(u, v) \in E$ no implica necesariamente que exista la conexión inversa. En este caso, el par (u, v) representa el enlace desde u hasta v, que gráficamente se muestra con una flecha de u a v. ▪ Una manera de representar a un grafo es a partir de su matriz de adyacencia $\mathbf{A} \in \mathrm{R}^{|V|\times|V|}$, que se define como $A_{ij} = 1$ si $(i, j) \in E$, y $A_{ij} = 0$ caso contrario. Nótese que si el grafo es no dirigido, entonces la matriz de adyacencia es simétrica. Por el contrario, si el grafo es dirigido, la matriz de adyacencia no es simétrica.
- Dado un grafo, si cada uno de sus nodos tiene d características, se define la matriz de características (*features*) como $\mathbf{X} \in \mathrm{R}^{|V|\times d}$, donde la i−ésima fila de $\mathbf{X}$ contiene las d características del nodo $i \in V$.

2.2 Incrustaciones y mensajes neuronales

La característica que define a una GNN es que utiliza una forma de "paso de mensajes neuronales" en la que vectores de mensajes se intercambian entre nodos y se actualizan mediante redes neuronales. Para esto se generan incrustaciones o *embeddings* ocultos para cada nodo $u \in V$, a partir del grafo $G = (V,E)$ y la matriz de características de sus nodos X [3].

Así, durante cada paso de mensajes en una GNN, cada *embedding* oculto $\mathbf{h}_u^{(k)}$ correspondiente a cada nodo $u \in V$ es actualizado teniendo en cuenta la información acumulada a partir de los vecinos de *u,* es decir de su entorno $N(u)$. Dicha actualización está dada por:

$$\begin{aligned}\mathbf{h}_u^{(k+1)} &= \text{UPDATE}^{(k)}\left(\mathbf{h}_u^{(k)}, \text{AGGREGATE}^{(k)}\left(\{\mathbf{h}_v^{(k)} \mid v \in N(u)\}\right)\right) \\ &= \text{UPDATE}^{(k)}\left(\mathbf{h}_u^{(k)}, \mathbf{m}_{N(u)}^{(k)}\right),\end{aligned}$$

donde UPDATE y AGGREGATE son funciones diferenciables, $\mathrm{m}_{N(u)}$ es el "mensaje" que se recopila de los vecinos de *u,* y *k* indica la iteración o capa de la GNN (que para $k = 0$ se tiene que $\mathrm{h}_u^{(0)} = x_u$, es decir que se utilizan las características de cada nodo). Es decir, en cada iteración la incrustación se actualiza teniendo en cuenta el mensaje recibido por parte de su entorno y la incrustación en la capa anterior. Nótese además que, dado que la función AGGREGATE toma un conjunto como argumento, una GNN definida de esta manera es equivariante a permutaciones por diseño [3].

En el caso particular de las redes GCN, las funciones UPDATE y AGGREGATE son sumas. Además, se aplica una normalización diferente dependiendo de si se trata de un grafo dirigido o uno no dirigido:

$$\mathbf{m}_{N(u)} = \begin{cases} \sum_{v \in N(u) \cup \{u\}} \frac{\mathbf{h}_v}{\sqrt{|N(u)||N(v)|}} & \text{si el grafo es no dirigido} \\ \sum_{v \in N(u) \cup \{u\}} \frac{\mathbf{h}_v}{|N(u)|} & \text{si el grafo es dirigido} \end{cases}$$

Así, una GCN define su actualización como:

$$\mathbf{h}_u^{(k+1)} = \sigma\left(\mathbf{W}^{(k)} \mathbf{m}_{N(u)}^{(k)}\right), \ \forall u \in V, \tag{1}$$

donde W es una matriz de pesos (parámetros) que aprende la red, y σ es una función no lineal, por lo general σ = ReLU.

Por otro lado, aprovechando la notación matricial y sus ventajas a la hora de programar, se puede redefinir la ecuación 1 para actualizar todos los vértices a la vez:

$$\mathbf{H}^{(k+1)} = \begin{cases} \sigma\left(\bar{\mathbf{D}}^{-1/2} \bar{\mathbf{A}} \bar{\mathbf{D}}^{-1/2} \mathbf{H}^{(k)} \mathbf{W}^{(k)}\right) & \text{si el grafo es no dirigido} \\ \sigma\left(\bar{\mathbf{D}}^{-1} \bar{\mathbf{A}} \mathbf{H}^{(k)} \mathbf{W}^{(k)}\right) & \text{si el grafo es dirigido} \end{cases}, \tag{2}$$

donde $\bar{\mathbf{A}} = \mathbf{A} + \mathbf{I}$, con $\mathbf{A}$ la matriz de adyacencia e $\mathbf{I}$ la matriz identidad, y donde $\bar{\mathbf{D}}$ es la matriz (diagonal) de grados dada por $\bar{D}_{ii} = \sum_j \bar{A}_{ij}$.

2.3 Hipótesis a poner a prueba

Teniendo en cuenta lo anterior, se tienen las siguientes hipótesis que se pondrán a prueba:

1. La GCN puede aprender a discriminar entre clases cuando están balanceadas y cuando existe una buena "separación" entre los nodos pertenecientes a cada una.
2. Si la topología del grafo es tal que las clases están lo suficientemente "separadas", entonces la GCN puede aprender a discriminar entre clases independientemente del valor de las características de los nodos.
3. El aprendizaje de la GCN se puede ver afectado por el grado de desbalance entre clases.

3. Materiales y métodos

En esta sección se describen el hardware y software utilizado, y el detalle de los experimentos.

3.1 Descripción de hardware y software utilizado

Para llevar a cabo los experimentos aquí planteados, se utilizó un procesador AMD FX(tm)-6300 Six-Core con sistema operativo Ubuntu 20.04.6 LTS, disponible en la Facultad de Ingeniería de la Universidad Nacional de Cuyo, que cuenta con una placa de video NVIDIA GeForce GTX 1080 Ti de 12 GB.

Usando la versión 3.8.10 de python, se utilizaron las siguientes bibliotecas principales: (a) `tensorflow`: versión 2.9.3; (b) `spektral` [2]: biblioteca para el aprendizaje profundo de grafos, basada en la API Keras y TensorFlow 2. Versión 1.2.0; (c) `pandas`: versión 1.5.3; (d) `numpy`: versión 1.23.4; (e) `matplotlib`: versión 3.6.2.

3.2 Descripción de los experimentos realizados

Para comenzar, las bases de datos que se utilizaron en cada prueba fueron construidas para tal fin, es decir que se crearon grafos sintéticos modificando alguna particularidad específica dependiendo del experimento. En esta sección se describe el proceso de armado de grafos sintéticos, así como el pre-procesamiento llevado a cabo sobre los datos y la estructura de la red, y las métricas utilizadas para evaluar el desempeño de la GCN.

3.2.1 Grafos sintéticos

Para llevar a cabo cada una de las pruebas, en primer lugar, se crearon diferentes conjuntos de grafos dirigidos sintéticos, siguiendo lo propuesto en [1], esto es usando el algoritmo *Contextual Stochastic Block Model* (CSBM), que genera tanto características para los nodos así como el grafo. Cada dataset creado consta de diez

grafos, cada uno de los cuales tiene 100 nodos[1]. Para poder construir dichos grafos es necesario definir:

- Número de clases: 2 (dos), por tratarse de un problema de clasificación binaria. A cada nodo se le asigna una etiqueta que lo identifica con la clase correspondiente.
- Número de características que tendrán los nodos: 2 (dos). Se eligi´o esta cantidad por simplicidad.
- Valor medio de las características por clase.
- Probabilidad de existencia de enlaces entre nodos de diferentes clases.

A la hora de definir los diferentes conjuntos de grafos, se modificaron la media de las características por clase, y la probabilidad de conexión entre los nodos, como muestra la Tabla 1, consiguiendo así diferentes "niveles de separación" entre las clases. Por un lado, con los tres tipos de separación, se armaron datasets donde las clases estaban balanceadas entre sí. Por otro lado, únicamente con la configuración de los casos "separados" y "mezclados" se crearon grafos donde las clases no estaban balanceadas, y la clase minoritaria representaba los siguientes porcentajes sobre el total: $1, 2, 5, 10, 20$. Algunos ejemplos de estos grafos sintéticos se observan en las Figuras 1 y 2.

Tabla 1. Niveles de separación entre los nodos pertenecientes a diferentes clases.

Caso	Probabilidad de conexión	Media de características por clase
"separados"	$\begin{bmatrix} 0{,}8 & 0{,}2 \\ 0{,}3 & 0{,}7 \end{bmatrix}$	$\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$
"juntos"	$\begin{bmatrix} 0{,}5 & 0{,}5 \\ 0{,}5 & 0{,}5 \end{bmatrix}$	$\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$
"mezclados"	$\begin{bmatrix} 0{,}5 & 0{,}5 \\ 0{,}5 & 0{,}5 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$

A su vez, en los casos de clases balanceadas se puso a prueba el comportamiento de la GCN al aplanar todas las *features* de los nodos a un valor constante, en este caso, 1 (uno). Estas pruebas tienen la particularidad de que los grafos sintéticos generados son iguales, desde un punto de vista topológico, a los generados anteriormente, es decir, los nodos guardan la misma proporción entre clases, presentan el mismo "nivel de separación" y tienen las mismas conexiones. La única diferencia es que al leer la información de las características, en lugar de las *features* generadas, asigna a cada nodo un mismo valor.

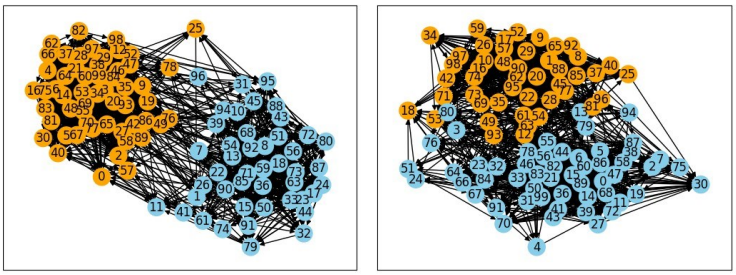
Una vez que se cargan los grafos en memoria, y antes de realizar cada prueba, se realiza un pre-procesamiento a los datos que está descripto en la próxima sección.

[1] La elección en el número de nodos radica en minimizar los tiempos de entrenamiento de la GCN.

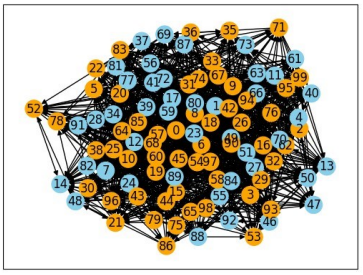
3.2.2 Pre-procesamiento y estructura de la red

Cada prueba cuenta con su propio conjunto de grafos como datos de entrada. A su vez, por cada grafo se tienen tres "componentes": la matriz de adyacencia, la matriz de características de los nodos y el vector de etiquetas de los nodos, usualmente representado como una matriz one-hot.

Antes de ingresar los datos a la red neuronal, se realiza un pre-procesamiento a la matriz de adyacencia $\mathbf{A}$ y a la matriz de características de los nodos $\mathbf{X}$. Por un lado, como se explica en la sección 2.2, se calcula $\bar{\mathbf{A}} = \mathbf{A} + \mathbf{I}$, donde $\mathbf{I}$ es



(a) "Separados" (b) "Juntos"



(c) "Mezclados"

Figura1. Ejemplos de grafos con clases balanceadas.

la matriz identidad, para luego normalizar $\bar{\mathbf{A}}$. Dicha normalización depende de si la matriz de adyacencia es o no simétrica. En este caso, como los grafos son dirigidos, siguiendo la ecuación 2, la normalización se calcula como $\bar{\mathbf{D}}^{-1}\ \bar{\mathbf{A}}$. Por otro lado, la matriz de características se normaliza por filas, es decir, para cada fila de $\mathbf{X}$ se divide cada entrada por la suma de los elementos de dicha fila.

Durante el entrenamiento se usó una tasa de aprendizaje (*learning rate*) de 0,01 y el optimizador Adam. En cuanto a la estructura de la red, se tiene:

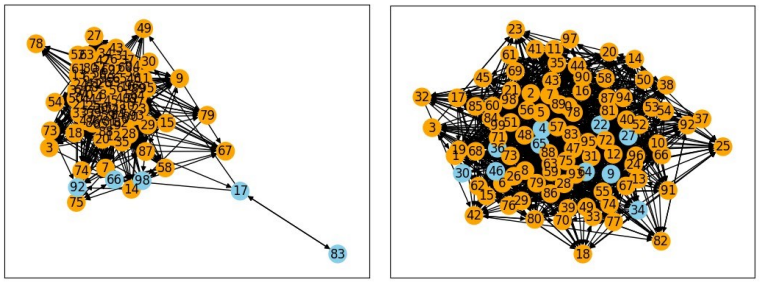
1. Dropout: rate = 0,5.
2. Capa GCN intermedia: 16 neuronas y función de activación ReLU.
3. Dropout: rate = 0,5.
4. Capa GCN final: 2 neuronas y función de activación Softmax.

A la hora de entrenar la red, se optó por un esquema k–fold, con $k = 10$, esto es para cada prueba se entrenaron diez modelos, dejando un grafo diferente para testeo cada

vez. Además, para evaluar el desempeño de la GCN, se observaron ciertas métricas que se describen a continuación.

3.2.3 Métricas utilizadas

En un problema de clasificación binaria se puede evaluar el desempeño del modelo a partir de varias métricas. Se toma como referencia la matriz de confusión que puede construirse a partir de la comparación entre las etiquetas reales y las predicciones hechas por la red, y se utilizan las siguientes dos métricas para evaluar el desempeño de la GCN:



(a) "Separados" (b) "Mezclados"

Figura2. Ejemplos de grafos con clases no balanceadas. A la izquierda, la clase minoritaria es del 5%, mientras que a la derecha es del 10%.

- AUC (Área bajo la curva ROC): la curva ROC muestra la relación de la tasa de VP contra la tasa de FP.
- Especificidad: proporción de predicciones correctas de clase 1, dado por VN/(VN+FP).

Vale aclarar que no se tomaron las métricas exactitud (*accuracy*) y precisión para evaluar las predicciones, ya que no siempre son representativos, especialmente cuando se tienen conjuntos de datos con clases no balanceadas. Por otro lado, como el interés está puesto principalmente en la clase 1, que coincide con la clase minoritaria en los casos no balanceados, se tomó la especificidad (*specificity*) y no su análogo con la clase positiva, conocido como *recall*.

Una vez descriptos los tipos de grafos, la red y las métricas a evaluar, se presentan ahora los principales resultados obtenidos.

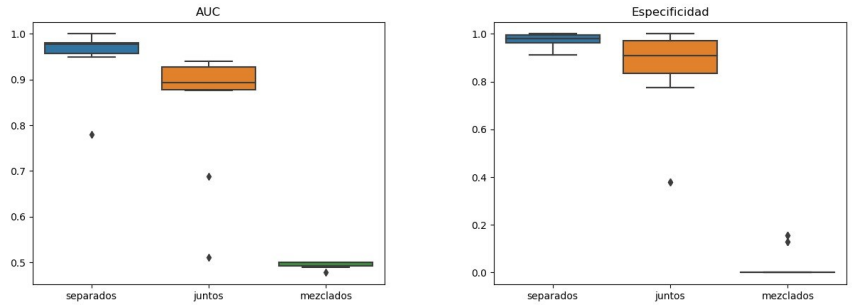
4. Resultados y discusión

Se comienza entrenando una red neuronal GCN, usando los mismos hiperparámetros descriptos en la sección anterior, pero variando los conjuntos de entrenamiento, para poner a prueba cada una de las hipótesis formuladas.

4.1 Comportamiento según el nivel de separación entre clases

Iniciando por los grafos con clases balanceadas, se pone a prueba la GCN variando el "nivel de separación" entre clases, usando las *features* de los nodos de cada grafo sin modificar. Como de cada experimento se corrieron diez entrenamientos variando en cada caso el conjunto de testeo, se muestran en la Figura 3 las gráficas boxplot, realizadas a partir de las predicciones sobre los diferentes conjuntos de testeo, de los valores AUC (a) y especificidad (b).

Como se esperaba, se observa que la GCN tiene un mejor comportamiento cuando los nodos pertenecen a clases separables, logrando valores de AUC y especificidad mayores a $0,95$, mientras que cuando se empiezan a juntar las clases baja el rendimiento de la red a $\sim 0,8$, y por el contrario, cuando los nodos están completamente mezclados la GCN no logra identificarlos correctamente, obteniendo entonces un AUC promedio de $0,5$ y una especificidad de 0.



(a) (b)

Figura3. Valores de AUC y especificidad obtenidos sobre las predicciones realizadas en los diferentes conjuntos de testeo, para diferentes niveles de separación entre clases. Nótese que la escala del eje de las ordenadas de cada grafica es diferente.

4.2 Comportamiento al aplanar las características de los nodos

En segundo lugar se puso a prueba el comportamiento de la red al aplanar las características de los nodos a un mismo valor constante, en este caso a 1, pero manteniendo el nivel de separación de las clases y la topología de los grafos utilizados en los experimentos de la sección anterior.

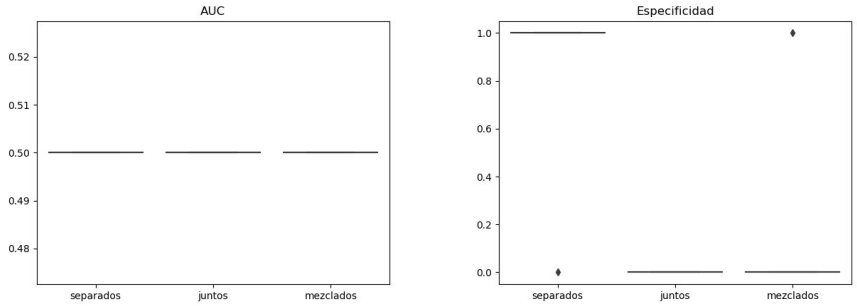
Como se puede observar en la Figura 4, al forzar que todas las características de los nodos de los grafos tengan el mismo valor constante, se pierde la capacidad de la red de diferenciar entre clases. Es decir, a pesar de lo que se esperaba, la topología de los grafos usados durante el entrenamiento no es suficiente para que la GCN logre diferenciar entre nodos. Se observa en el gráfico (a) que el valor de AUC fue $0,5$ para cada una de las corridas, mientras que el gráfico (b) se aprecia que la especificidad fue de 0 o 1, lo que indica que la GCN predijo la totalidad de los nodos como pertenecientes a sólo una de las clases.

4.3 Comportamiento ante clases no balanceadas

Por último, cuando se tienen clases que no están balanceadas, la red tiene un rendimiento mucho menor. Se puede observar en la Figura 5 que cuando la clase minoritaria representa un porcentaje menor al 5% del total de nodos, la GCN clasifica a todos los nodos como pertenecientes a una única clase.

Sin embargo, al aumentar el porcentaje de la clase minoritaria la red comienza a aprender a diferenciar entre clases, aunque el desempeño logrado es menor al caso balanceado.

Este comportamiento se observa para los grafos construidos a partir de la configuración "separados". Por otro lado, de manera análoga a los resultados de las secciones anteriores, para la configuración "mezclados" la red no logra aprender y clasifica a todos los nodos como una única clase. Estas graficas no se muestran por ser similares a las de la Figura 4.



(a) (b)

Figura4. Valores de AUC y especificidad obtenidos sobre las predicciones realizadas en los diferentes conjuntos de testeo, para diferentes niveles de separación entre clases, al entrenar aplanando las características de los nodos a un mismo valor constante. Nótese que la escala del eje de las ordenadas de cada grafica es diferente.

5. Conclusiones y trabajo futuro

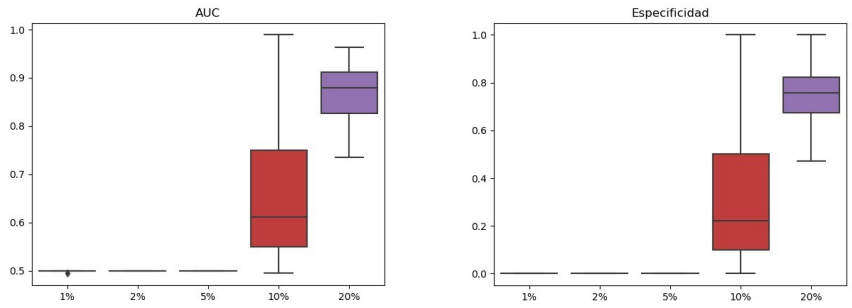
En este trabajo se realizó una exploración de las redes GCN, para analizar c´omo funcionan y ver su comportamiento ante diferentes escenarios, dados por distintas configuraciones de grafos.

Como se esperaba, cuando la topología del grafo es tal que las clases son separables entre sí, la GCN logra un muy buen rendimiento, aprovechando no sólo las características de cada nodo, sino también la información aportada por los vecinos. A su vez, ante problemas con una marcada desproporción entre clases, la red no logra distinguir la clase minoritaria. Sin embargo, este comportamiento comienza a mejorar a medida que aumenta el porcentaje de nodos en dicha clase. Por otro lado, a pesar de lo esperado, la topología del grafo no es información suficiente para la GCN.

A partir de lo anterior se concluye que, a la hora de entrenar una red GCN, se verifique en primer lugar que los nodos de los grafos tengan características lo suficientemente "buenas" para obtener mejores resultados.

Como trabajo a futuro se proponen diferentes pruebas que complementarían este trabajo, a fin de alcanzar un mayor entendimiento de este tipo de redes, y que no se llevaron a cabo por limitación en el tiempo. Algunas ideas de pruebas a realizar:

- Ante casos de *features* uniformes, evaluar si el considerar características como el grado (u otras relativas a la estructura del grafo) de los nodos agrega información relevante, y mejora el desempeño de la red.
- Poner a prueba diferentes estrategias de aumento de datos, o función de pérdida ponderada, para evaluar el comportamiento de la red en casos de clases no balanceadas.
- Probar modificaciones en la estructura de la GCN, así como generar grafos con otros tipos de topologías.



(a) (b)

Figura5. Valores de AUC y especificidad obtenidos sobre las predicciones realizadas en los diferentes conjuntos de testeo, para grafos con clases no balanceadas. Se probaron diferentes configuraciones, donde la clase minoritaria (la clase de interés) representa el 1,2,5,10 o 20 por ciento del total de nodos. Nótese que la escala del eje de las ordenadas de cada gráfica es diferente.

Referencias

1. Carson, B.S.: A survey of graph neural networks on synthetic data (2023)
2. Grattarola, D., Alippi, C.: Graph neural networks in tensorflow and keras with spektral [application notes]. IEEE Computational Intelligence Magazine 16(1), 99– 106 (2021)
3. Hamilton, W.L.: Graph representation learning. Synthesis Lectures on Artifical Intelligence and Machine Learning 14(3), 1–159 (2020)
4. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Aplicación de Procesos de Diseño de Bases de Datos NoSQL

Luciano Marrero[1], Verena Olsowy[1] [0000-0002-9003-8170], Fernando Tesone[1] [0000-0001-9499-3127],
Pablo Thomas[1] [0000-0001-9861-987X], Patricia Pesado[1] [0000-0003-0000-3482]

[1] Instituto de Investigación en Informática LIDI
Facultad de Informática - Universidad Nacional de La Plata – Argentina
Centro Asociado Comisión de Investigaciones Científicas de la Provincia de Buenos Aires
{lmarrero, volsowy, ftesone, pthomas, ppesado}@lidi.info.unlp.edu.ar

Resumen: Para las bases de datos relacionales, es deseable realizar un proceso de diseño que inicie con un modelo conceptual, luego se derive a un modelo lógico relacional y finalice con un modelo físico de datos. En el caso de las bases de datos NoSQL, no poseen un proceso masivamente aceptado para el diseño de esquemas de datos, sino que han surgido como soluciones directamente a nivel físico. No obstante, a medida que estas bases de datos han adquirido popularidad, se han propuesto diferentes procesos y/o metodologías de diseño NoSQL, necesarios para comprender la semántica de los datos. En este trabajo se plantea la utilización y aplicación de procesos de diseño de bases de datos NoSQL a un caso de estudio real. El principal objetivo es evaluar la viabilidad de los procesos estudiados en un contexto determinado.

Keywords: Base de datos NoSQL, procesos de diseño de bases de datos, modelado de datos en bases de datos NoSQL, almacenamiento no estructurado de datos.

1. Introducción

Las metodologías de diseño y construcción de bases de datos relacionales han sido estudiadas y aplicadas por décadas. Inicialmente, se presenta un esquema conceptual de alto nivel del problema a resolver. Posteriormente, se realiza un mapeo del esquema conceptual al esquema lógico relacional y finalmente, se obtiene una representación del esquema físico de la base de datos. Los principios o reglas que se aplican a un esquema de datos relacional no resultan apropiados para una base de datos no relacional. Esto se debe a que estas bases de datos representan un conjunto de variadas implementaciones a nivel físico en donde cada una de ellas posee su propia forma de estructurar los datos. En general, NoSQL, se basa en la redundancia, la desnormalización y las consultas que se impactarán sobre el sistema. Las Bases de Datos relacionales y NoSQL proponen modelos diferentes, por lo tanto, su proceso de diseño también tendrá que ser diferente [11, 12].

En este trabajo se propone la aplicación de dos procesos de diseño diferentes para bases de datos NoSQL a un caso de estudio real con el objetivo de realizar un análisis comparativo sobre las características principales de cada proceso utilizado.

A partir de la sección 2, el trabajo se organiza del siguiente modo: se describen las características principales de las Bases de Datos NoSQL, en la sección 3 se presentan

los procesos de diseño seleccionados, en la sección 4 se aplican dichos procesos a un caso de estudio real, en la sección 5 se presenta un análisis de resultados y finalmente, en la sección 6 y 7 se expresan las conclusiones generales y trabajos futuros.

2. Características de las Bases de Datos NoSQL

Las Bases de Datos NoSQL representan un conjunto de tipos de Bases de Datos que poseen sus propias implementaciones para el almacenamiento de datos. NoSQL se distingue de los tradicionales sistemas de gestión de Bases de Datos relacionales en diversos aspectos; no poseen un lenguaje de consulta estructurado (SQL) como lenguaje principal, no requieren de una estructura fija y tabular, no soportan operaciones JOIN, no garantizan por completo las propiedades de ACID (atomicidad, consistencia, aislamiento y durabilidad), y en general, su estructura se basa en la escalabilidad horizontal [11, 13, 14].

NoSQL propone un sistema llamado "BASE (Básicamente Disponible, Estado Suave, Consistencia Eventual)". A través de estas propiedades se logra disponibilidad básica (Base Availability), esto significa que el sistema se encontrará disponible la mayoría del tiempo. Con el estado débil (Soft State) el sistema se vuelve más flexible en cuanto a consistencia y con la consistencia eventual (Eventual Consistency) se garantiza que el sistema eventualmente se volverá consistente [12, 13, 14].

Dependiendo de la forma en que se almacene la información, existen cuatro categorías principales de almacenamiento para Bases de Datos NoSQL.

Almacenamiento Clave/Valor: simples en cuanto a su implementación, almacenan datos como un conjunto de pares "clave/valor" (key-value). La clave representa un identificador único que puede retornar un objeto complejo y arbitrario de información, denominado valor (value). Por ejemplo, Redis y Amazon DynamoDB, entre otros, implementan este tipo de almacenamiento [3, 4].

Almacenamiento Documental: el concepto central de este tipo de almacenamiento es el documento. Una Base de Datos NoSQL Documental, almacena, recupera y gestiona datos de documentos. Estos documentos encapsulan y codifican datos o información bajo algún formato estándar (XML, YAML, JSON, BSON). Por ejemplo, MongoDB y Apache CouchDB, entre otros, son implementaciones de Bases de Datos Documentales [5, 6].

Almacenamiento de Familia de Columnas: en este tipo de almacenamiento los datos se encuentran organizados por columnas, en lugar de filas. Las Bases de Datos que utilizan esta forma de almacenamiento tienden a ser un híbrido entre las Bases de Datos Relacionales y la tecnología orientada a columna. Por ejemplo, Cassandra y Apache HBase, entre otros, utilizan este tipo de almacenamiento [7, 8].

Almacenamiento de Grafos: en este tipo de almacenamiento se representa la Base de Datos bajo el concepto de un grafo, permitiendo almacenar la información como nodos y sus respectivas relaciones, con otros nodos, mediante aristas. Se aplica la teoría de grafos para recorrer la estructura. Son útiles para almacenar información en modelos que poseen numerosas relaciones entre sus datos. Neo4j y OrientDB, entre otros, implementan este tipo de almacenamiento [9, 10].

3. Procesos de diseño para bases de datos NoSQL utilizados

En los últimos años, han aparecido metodologías para el diseño de bases de datos NoSQL. En [15] se realizó una revisión bibliográfica sobre este tema. Como resultado de esta revisión en [16] se ha presentado un estudio comparativo de procesos para el diseño de bases de datos NoSQL. En este trabajo se plantea la aplicación de los siguientes procesos de diseño a un caso de estudio real:

- **Mortadelo**: Generación automática de almacenamiento NoSQL a partir de modelos de datos independientes de la plataforma[1]**:**

  Mortadelo se describe como un proceso de diseño de bases de datos NoSQL basado en modelos, en donde, a partir de un modelo conceptual de datos, independiente de cualquier tipo de bases de datos, se genera de forma autónoma una posible implementación para un motor de bases de datos NoSQL concreto. Además, este proceso permite configurar el diseño final según las necesidades de cada contexto.

  Este proceso necesita operar con modelos bien definidos, en particular, necesita como entrada un metamodelo en el cual se especifican el modelo conceptual de datos y las consultas que recuperarán y actualizarán la información representada. Además, permite realizar anotaciones generales, por ejemplo, la tasa de actualización que posiblemente tendrá una entidad.

- **NoAM** (Modelo abstracto NoSQL): Modelado de datos en el mundo NoSQL [2]:

  Se propone un proceso de diseño que tiene una fase conceptual, una fase lógica, la cual es independiente del tipo de base de datos, y una fase final que considera las características específicas de un motor de base de datos NoSQL. NoAM se basa en las siguientes actividades principales:

  A. Modelado conceptual de datos a partir del Diseño Dirigido al Dominio (DDD) que da como resultado un diagrama UML. No se menciona cómo realizar este diagrama.
  B. Sobre el diagrama UML del punto anterior se identifican los agregados. Un agregado es un grupo de objetos relacionados que representa una unidad de acceso y de manipulación atómica.
  C. Implementación del modelo NoAM a partir de la identificación de agregados y su traducción correspondiente al nivel físico.

4. Aplicación de los procesos de diseño para bases de datos NoSQL

Inicialmente, se presenta el esquema conceptual de datos que será utilizado como entrada para ambos procesos de diseño (Mortadelo [1] y NoAM [2]). Este esquema conceptual se basa en una plataforma educativa que es utilizada por los estudiantes de la Facultad de Informática de la UNLP. A continuación, en la figura 1 se presenta el esquema conceptual propuesto utilizando un Diagrama Entidad Relación (DER).

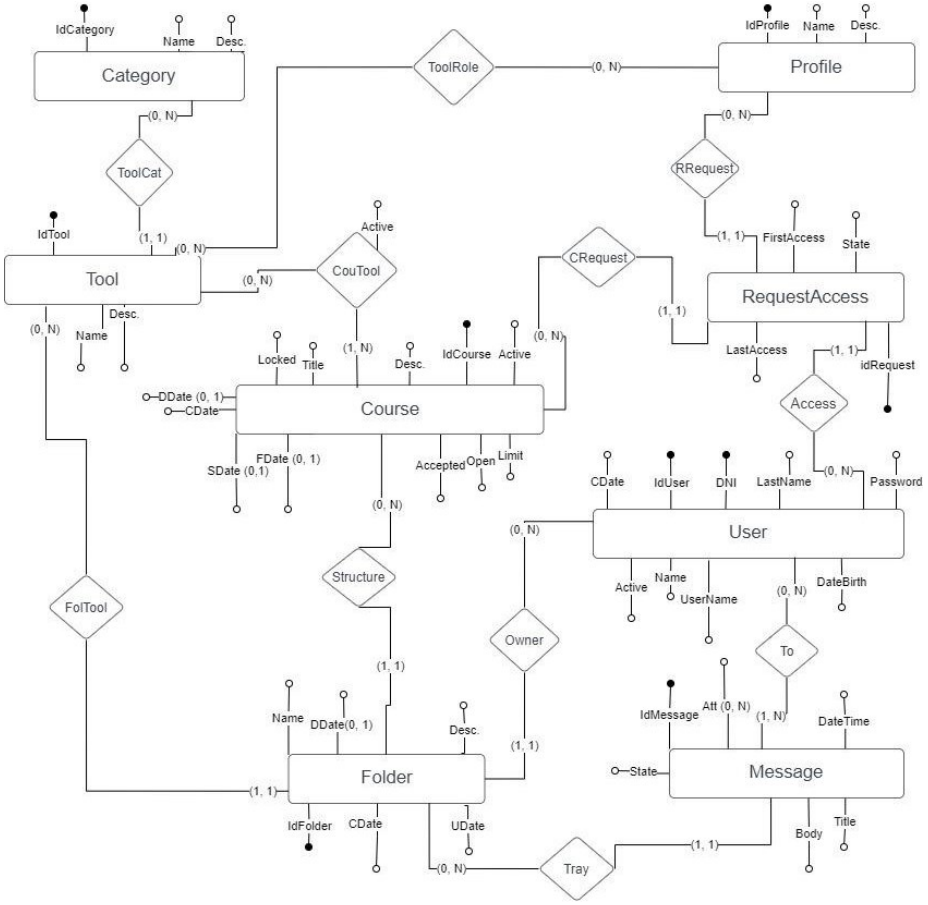


Figura 1.Esquema conceptual para el caso de estudio propuesto utilizando un DER.

La figura 1 representa una parte que corresponde a la administración de usuarios y cursos. Abarca las solicitudes de acceso a cada curso por parte de los usuarios, las herramientas disponibles según el perfil del usuario y la mensajería interna de cada curso. A continuación, se presenta la aplicación de ambos procesos de diseño (Mortadelo [1] y NoAM [2]) para el esquema conceptual propuesto.

4.1 Aplicación de Mortadelo

Mortadelo necesita operar con modelos bien definidos. A continuación, en la figura 2 se especifican los componentes generales que propone este proceso de diseño para una base de datos documental [1].

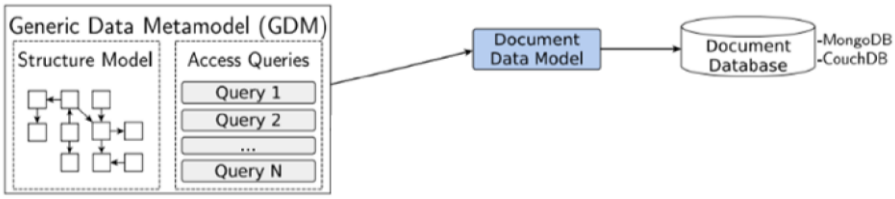


Figura 2. Componentes generales de Mortadelo para bases de datos documentales.

Inicialmente, se necesita definir el Metamodelo de Datos Genéricos (GDM) que incluye el modelo conceptual de datos y las consultas que impactarán en el sistema.

Para este caso de estudio, se definen 4 consultas que se presentan a continuación:

- **Consulta 1**: Listado de cursos activos para un usuario determinado.
- **Consulta 2**: Herramientas habilitadas para un curso determinado.
- **Consulta 3**: Mensajes de un usuario para un curso y carpeta determinada.
- **Consulta 4**: Para cada curso activo de un usuario, cantidad de mensajes aún no leídos en la bandeja de entrada.

En el siguiente paso, las entidades y consultas del modelo se pueden expresar en el lenguaje GDM (sintaxis textual para instanciar modelos conformes al GDM). A continuación, en la figura 3 se presentan las implementaciones en el lenguaje GDM para las entidades *Tool*, *User* y *Request* y las consultas 1, 2 y 3.

Entidades

```
entity Tool {
  id idTool unique
  text name
  text desc
  ref Category[1]: category
}
```

```
entity User {
  id idUser unique
  text dni unique
  text lastName
  …
}
```

```
entity Request {
  id idRequest unique
  number state
  datetime firstAccess
  datetime lastAccess
  ref User[1]: user
  ref Course[1]: course
  ref Profile[1]: profile
}
```

Consultas

```
query C1_coursesByUser:
  select course.title
  from Request as req
  including req.user as user,  req.course as course
  where req.state = "?" and user.userName = "?" and user.password = "?"
  order by course.title
```

```
query C2_toolsForIdCourse:
  select tool.name, cat.name
  from CourseTool as ct
  including ct.tool as tool, ct.course as course, ct.tool.category as cat
  where ct.active and course.idCourse = "?"
  order by tool.name
```

```
query C3_MessageForIdCourseIdFolderIdUser:
  select mess.title, mess.body, mess.datetime
  from Folder as fol
  including fol.course as course, fol.owner as owner
            fol.messages as mess
  where fol.idFolder = "?" and course.idCourse = "?" and owner.idUser = "?"
  order by mess.datetime
```

Figura 3. Representación de entidades y consultas en lenguaje GDM.

Luego, se debe realizar un modelo lógico de alto nivel de acuerdo al tipo de almacenamiento que implemente el motor de bases de datos NoSQL a utilizar. En este caso, se utilizará un motor de base de datos documental, por ejemplo, MongoDB. En la figura 4, se presenta el modelo lógico genérico que propone este proceso para un motor de base de datos documental.

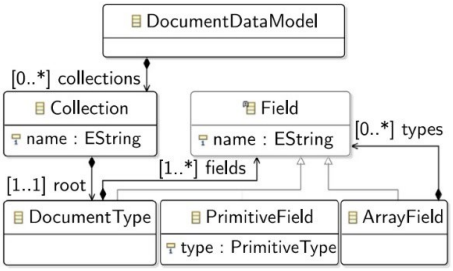


Figura 4. Modelo de nivel lógico para almacenamiento documental.

En general, los motores de bases de datos documentales almacenan documentos en un conjunto de colecciones que poseen un nombre que las identifica. Cada colección almacena documentos que, en la mayoría de los casos, comparten la misma estructura o tipo de documento (*DocumentType*). Un tipo de documento se define a través de un conjunto de campos, que pueden ser elementos primitivos, matrices de elementos o incluso tipos de documentos anidados. Todas las colecciones tienen un tipo de documento principal, que en el modelo lógico propuesto se denomina raíz (root).

Las bases de datos documentales permiten la creación de colecciones heterogéneas (documentos con diferentes esquemas). No obstante, este proceso de diseño propone dividir cualquier colección heterogénea en varias homogéneas, es decir, que cada colección se ajuste a un solo esquema [1].

En una base de datos documental, cuando una consulta solicita recuperar información, se intenta obtener desde una única colección, evitando la necesidad de generar operaciones innecesarias. Para ello, en este proceso se identifican los puntos de entrada al GDM, estos puntos se determinan según las entidades principales de las consultas planteadas, es decir, las entidades expuestas en la cláusula *from*. Por ejemplo: *Request* de la consulta 1, *CourseTool* de la consulta 2 y *Folder* de la consulta 3 y 4.

Posteriormente, siguiendo el modelo lógico de alto nivel presentado en la figura 4, para cada una de estas entidades se crea una colección y se definen los tipos de documentos raíz que contendrá cada colección correspondiente a la entidad principal seleccionada. A continuación, para cada consulta en donde aparecen las entidades principales seleccionadas en las cláusulas *from* (*Request, CourseTool y Folder*), se analizan las referencias incluidas (cláusula *including*). A partir de éstas referencias, para cada consulta, se construye un árbol de acceso que determina las entidades a las que se debe acceder para responder a la consulta correspondiente. En la figura 5 se presenta el árbol de acceso generado para la consulta 2.

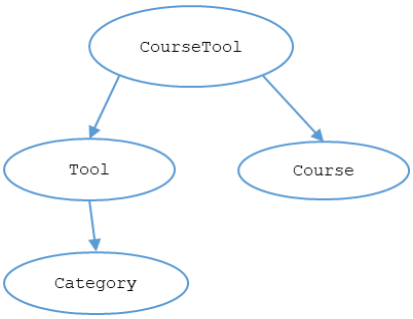


Figura 5. Árbol de accesos para la consulta 2.

Utilizando el árbol de acceso, se genera el contenido del tipo de documento raíz para la colección *CourseTool*. Inicialmente, este documento contendrá todos los atributos de la entidad raíz de la consulta 2, es decir, todos los atributos de la entidad *CourseTool*. Posteriormente, se procesan las referencias, en donde para cada una de ellas se genera un nuevo campo en el documento con la información correspondiente, es decir, que se genera un documento embebido o una colección de documentos dependiendo de la cardinalidad de la referencia. En la figura 6, se presenta el esquema físico en formato *JSON* para MongoDB derivado de la figura 5.

Para cada consulta definida, se deben aplicar los mismos pasos.

```
CourseTool {
  _id, active,
  tool: {name, desc, category: {name, desc}},
  course: {title, desc, startDate, finishDate, limit, accepted, …}
}
```

Figura 6. Parte del esquema físico generado para MongoDB.

4.2 Aplicación de NoAM.

NoAM propone un proceso de diseño que incluye una fase conceptual, seguida de una fase de diseño lógico independiente del motor de bases de datos NoSQL a utilizar y una fase final que tendrá en cuenta características del motor de base de datos NoSQL elegido.

Inicialmente, se propone organizar el conjunto de datos de una base de datos NoSQL en agregados. Cada agregado, es un grupo de objetos o entidades que representan una unidad de acceso y manipulación de datos. En la figura 7 se presentan algunos agregados que se pueden definir para el DER presentado en la figura 1.

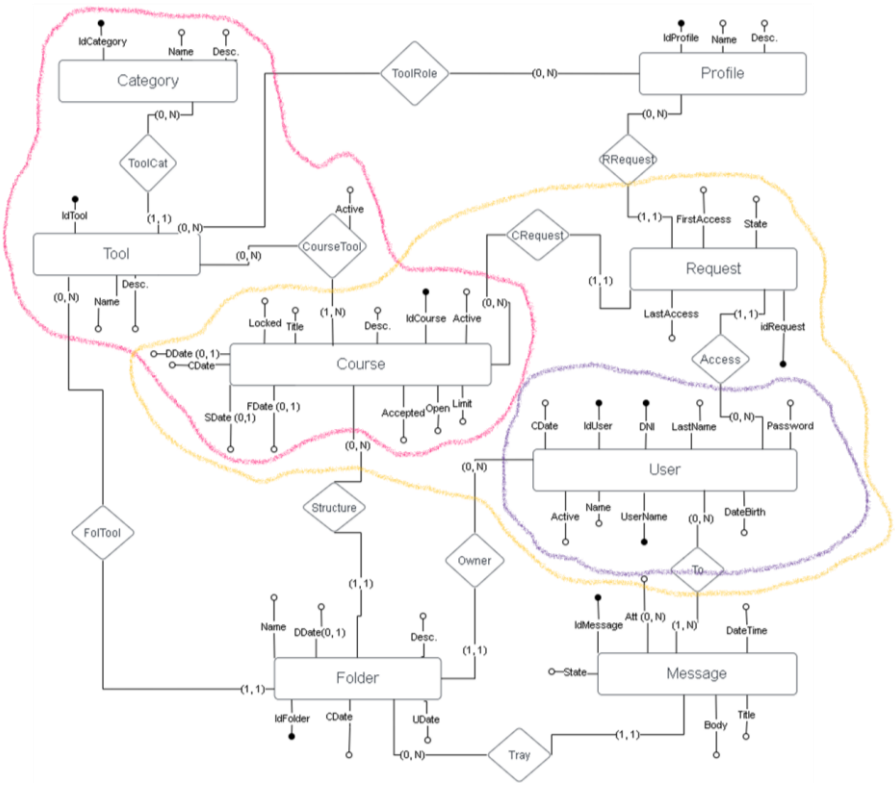


Figura 7. Ejemplo de agregados para el DER presentado en la figura 1.

Las bases de datos NoSQL poseen alternativas de almacenamiento diferentes, no obstante, a pesar de las diversas implementaciones, se pueden identificar algunas características comunes para representar los agregados definidos. En este caso se representarán los agregados para una base de datos documental, por ejemplo MongoDB.

En una base de datos documental, cada agregado se representará por un único documento principal y la colección que contendrá los documentos corresponde a una entidad o tipo de agregado. A continuación, en la figura 8, se presenta una posible representación de los agregados expuestos en la figura 7.

A partir de aquí, el proceso continúa aplicando el modelo de datos NoAM. Este modelo aprovecha la forma de acceso y distribución de los datos que poseen en general las bases de datos NoSQL. En la figura 9 y figura 10, se presentan fragmentos de una posible representación de una base de datos NoAM para los agregados de la figura 7.

A partir de la notación en NoAM, se puede obtener el esquema físico para una base de datos documental, por ejemplo, MongoDB. A continuación, en la figura 11, se presenta el esquema físico en formato JSON para MongoDB.

Para el resto de los agregados, se deben aplicar los mismos pasos.

Colección	Id del documento	Documento
User	27.001.002	{"_id":"27.001.002", "DNI":"27.001.002", "Name":"Luciano", …}
User	35.001.002	{"_id":"35.001.002", "DNI":"35.001.002", "Name":"Verena", …}
Request	12345	{"_id":"12345", "state":"accepted", "user":{…}, "course":{…}, …}
Request	12346	{"_id":"12346", "state":"rejected", "user":{…}, "course":{…}, …}
Course	10	{"_id":"10", "title":"NoSQL", "active":"true", "limit":"150", tools:[{…},{…},{…},…], …}

Figura 8. Ejemplo de agregados para una base de datos documental (MongoDB).

User

27.001.002	{ DNI: 27.001.002, lastName: "Marrero", name: "Luciano", userName: "lucho", password: "###!&&%%~~", … }
35.001.002	{ DNI: 35.001.002, lastName: "Olsowy", name: "Verena", userName: "vere", password: "#!!@@&&%##~", … }

Figura 9. Anotación para una parte de la base de datos en NoAM.

Request

12345	{ idRequest: 12345, state: "accepted", firstAccess: "2019-04-07", … user: { DNI: 35.001.0002, name: "Verena", …}, course: { title: "NoSQL", desc: "Curso NoSQL",…} }

Figura 10. Anotación para una parte de la base de datos en NoAM.

```
Request{ _id, state, firstAccess,… ,
  user: { DNI, name, …},
  course: { title, desc, …}
}
```

Figura 11. Parte del esquema físico generado para MongoDB.

5. Análisis de resultados

Luego de aplicar ambos procesos de diseño para bases de datos NoSQL a un caso de estudio real, se realiza un análisis comparativo con el objetivo de destacar aspectos a considerar en base al experimento realizado.

En primer lugar, se aplicó Mortadelo [1]. Este proceso se basa en la definición de un Metamodelo de Datos Genéricos (GDM), en el cual se considera el modelo

conceptual inicial y las consultas principales del sistema. Identificar estas consultas en una etapa inicial y en un modelo de datos complejo, puede ser una tarea compleja. Además, se debe considerar que si surge una nueva consulta en etapas avanzadas del diseño, implica realizar una revisión sobre los pasos seguidos a partir del GDM.

Por otro lado, en el caso de tener varias consultas, se puede generar un alto grado de desnormalización de los datos. Si bien el proceso expone esta situación y propone medidas para mitigarlas, no son triviales para el diseñador. No obstante, aunque requiere de bastante tiempo (según la experiencia del diseñador), con Mortadelo [1] se ha logrado generar un esquema físico completo para una base de datos NoSQL documental (MongoDB).

En segundo lugar, se aplicó NoAM [2]. En este proceso, una vez definido el modelo conceptual de datos, se deben identificar los agregados (unidades de acceso y manipulación de datos). Esta es una tarea subjetiva, se podrían generar tantos agregados como entidades o clases posea el modelo, o por el contrario, generar muy pocos agregados. Este primer paso, requiere conocer aspectos del modo de acceso y manipulación de los datos que posee el motor de base de datos NoSQL de destino. Además, para definir correctamente los agregados, es importante identificar las consultas principales que impactarán contra el modelo de datos.

Luego, el proceso continúa aplicando el modelo de datos NoAM. Este modelo plantea como método de almacenamiento central de datos la implementación de colecciones, independientemente del motor de base de datos NoSQL de destino. Además, la notación final que se brinda es genérica y puede ser interpretada por el diseñador de diferentes maneras según su experiencia con el motor de base de datos NoSQL a utilizar. No obstante, a través de NoAM [2], se ha logrado generar un esquema físico completo para una base de datos NoSQL documental (MongoDB) sin grandes complicaciones, pero al igual que Mortadelo [1] requiere tiempo, bastante conocimiento y experiencia en diseño de bases de datos en general.

6. Conclusiones

Este trabajo se centra en la aplicación de dos procesos o metodologías de diseño para bases de datos NoSQL, Mortadelo [1] y NoAM [2], a un caso de estudio real. Ambos procesos han sido seleccionados a través de una revisión bibliográfica exhaustiva realizada sobre el tema en [15].

A través de este experimento, se pudo observar en ambos casos que a partir de un esquema conceptual de datos se logró generar un esquema físico para un motor de base de datos NoSQL documental, en este caso MongoDB [5]. No obstante, ambos procesos poseen similitudes y diferencias que se pueden distinguir al momento de su aplicación.

Tanto Mortadelo [1] como NoAM [2] inicialmente necesitan un conjunto de actividades que requieren un conocimiento preciso del dominio del problema a resolver. Los cambios de requerimientos en etapas avanzadas, pueden implicar una revisión completa de algunas decisiones y acciones aplicadas sobre el esquema inicial.

Por otro lado, ambos procesos en la etapa de diseño lógico presentan ambigüedades y dependen de la experiencia del diseñador. En general, no se obtiene un esquema lógico explícito, lo que puede conducir a distintas variantes para el modelo físico. No obstante, en este punto, Mortadelo [1], presenta reglas mejor definidas que NoAM [2].

En lo que respecta al esquema físico final, Mortadelo [1] es más preciso que NoAM [2].

Finalmente, se comprobó que a través de ambos procesos se obtiene una representación posible del esquema físico final para una base de datos documental, por ejemplo MongoDB.

7. Trabajo Futuro

Como trabajo futuro, se prevé extender la aplicación de éstos procesos a otros tipos de bases de datos NoSQL, como por ejemplo, Familia de Columnas y Clave-Valor.

Además, se pretende integrar procesos de ingeniería inversa para bases de datos NoSQL y contrastar con los esquemas generados por Mortadelo [1] y NoAM [2].

Por otro lado, se buscará estudiar y añadir nuevos procesos de diseño para bases de datos NoSQL, especialmente para las orientadas a grafos, ya que los procesos aplicados no tienen en cuenta este tipo de base de datos NoSQL.

Bibliografía

1. Alfonso de la Vega, DiegoGarcía,Saiz Carlos Blanco,Marta Zorrilla, Pablo Sánchez. Mortadelo: Automatic generation of NoSQL stores from platform-independent data models.Future Generation Computer Systems. Volume 105, April 2020, Pages 455-474.
2. Paolo Atzeni, Francesca Bugiotti, Luca Cabibbo, Riccardo Torlone. Data Modeling in the NoSQL World. HAL open science. https://hal.archives-ouvertes.fr/hal-01611628.
3. Redis. https://redis.io/. Julio de 2023.
4. Amazon DynamoDB. https://aws.amazon.com/es/dynamodb/. Julio de 2023.
5. MongoDB. https://www.mongodb.com/es. Julio de 2023.
6. CouchDB. https://couchdb.apache.org/. Julio de 2023.
7. Apache Cassandra. https://cassandra.apache.org/_/index.html. Julio de 2023.
8. Apache HBase. https://hbase.apache.org/. Julio de 2023.
9. Neo4j. https://neo4j.com/. Julio de 2023.
10. OrientDB. https://orientdb.org/. Julio de 2023.
11. Análisis de performance en Bases de Datos NoSQL y Bases de Datos Relacionales. Pesado P., Thomas P., Delía L., Marrero L., Olsowy V., Tesone F.. XXVI Congreso Argentino de Ciencias de la Computación (CACIC 2020). ISBN 978-987-4417-90-9. http://sedici.unlp.edu.ar/handle/10915/114202.
12. Un estudio comparativo de bases de datos relacionales y bases de datos NoSQL. Pesado P., Thomas P., Delía L., Marrero L., Olsowy V., Tesone F., Fernandez S. J. XXV Congreso

Argentino de Ciencias de la Computación (CACIC 2019). ISBN 978-987-688-377-1. http://sedici.unlp.edu.ar/handle/10915/91403.

13. Aspectos de Ingeniería de Software, Bases de Datos Relacionales, y Bases de Datos No Relacionales y Bases de Datos Como Servicios en la Nube para el desarrollo de Software Híbrido. XXIII Workshop de Investigadores en Ciencias de la Computación (WICC 2021). ISBN 978-987-24611-3-3; 978-987-24611-4-0. http://sedici.unlp.edu.ar/handle/10915/120139.
14. NoSQL: modelos de datos y sistemas de gestión de bases de datos. XX Workshop de Investigadores en Ciencias de la Computación (WICC 2018, Universidad Nacional del Nordeste). http://sedici.unlp.edu.ar/handle/10915/67258.
15. Un estudio de procesos de diseño de bases de datos NoSQL. Marrero, L.; Olsowy, V.; Tesone, F.; Thomas, P.; Corbalán, L.; Fernández Sosa, J.; Pesado, P. XXVIII Congreso Argentino de Ciencias de la Computación (CACIC 2022). ISBN: 978-987-1364-31-2. http://sedici.unlp.edu.ar/handle/10915/149452.
16. NoSQL database design processes: a comparative study. Marrero, L.; Olsowy, V.; Thomas, P. XI Jornadas de Cloud Computing, Big Data & Emerging Topics. ISBN: 978-950-34-2271-7. http://sedici.unlp.edu.ar/handle/10915/155446.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Primeras Experiencias en la Identificación de Personas con Riesgo de Diabetes en la Población Argentina utilizando Técnicas de Aprendizaje Automático

Enzo Rucci[1], Gonzalo Tittarelli[2], Franco Ronchetti[1] , Jorge F. Elgart[3], Laura Lanzarini[1], and Juan José Gagliardino[3]

III-LIDI, Facultad de Informática, UNLP – CIC. La Plata (1900), Bs As, Argentina
{erucci,fronchetti,laural}@lidi.info.unlp.edu.ar
Facultad de Informática, UNLP. La Plata (1900), Bs As, Argentina
CENEXA, Facultad de C. Médicas, UNLP-CONICET. La Plata (1900), Bs As, Argentina
{jelgart,jjgagliardino}@cenexa.org

Resumen La detección de Diabetes Tipo 2 (DT2) y prediabetes (PDM) representa un verdadero desafío para la medicina debido a la ausencia de síntomas patogenómicos y a la falta de conocimiento de los factores de riesgo asociados. Si bien existen algunas propuestas de modelos de aprendizaje automático que permiten identificar a personas en riesgo, las características de esta enfermedad hacen que uno que resulte adecuado para una población, no necesariamente lo sea para otra. Este artículo propone desarrollar y evaluar modelos predictivos que permitan identificar personas con riesgo de DT2 y PDM específicos para la población argentina. Partiendo de un cuidadoso preprocesamiento de la base de datos, se generaron dos datasets particulares considerando el compromiso entre cantidad de registros y de variables disponibles. Luego de aplicar 5 diferentes modelos de clasificación, los resultados obtenidos muestran que algunos de ellos obtuvieron muy buenos rendimientos para ambos datasets. En particular, RF, DT y ANN demostraron gran poder de clasificación, con altos valores en las métricas consideradas. Considerando la vacancia de herramientas de este tipo para la población argentina, este trabajo representa el primer paso hacia modelos más sofisticados.

Keywords: salud pública · enfermedad crónica · machine learning

1. Introducción

La Diabetes Tipo 2 (DT2) es una enfermedad crónica caracterizada por niveles elevados de glucemia que se manifiesta cuando el páncreas endocrino es incapaz de producir la cantidad de insulina suficiente que requieren sus tejidos [11]. Debido a su creciente prevalencia en combinación con su elevado costo de atención [4,29], constituye un serio problema de salud pública, por lo que se han realizado grandes esfuerzos por desarrollar estrategias efectivas para su prevención y tratamiento oportuno, así como para evitar el desarrollo de sus complicaciones crónicas.

Resulta importante reconocer que las consecuencias negativas de esta enfermedad comienzan en una etapa previa conocida como prediabetes (PDM), la cual está definida como una elevación en la concentración de glucosa en sangre más allá de los niveles normales pero sin alcanzar los valores diagnósticos de diabetes. La PDM se manifiesta a través de la Glucemia en Ayunas Alterada (GAA), la Tolerancia a la Glucemia

Alterada (TGA) y la combinación de ambas [1]. La PDM implica un riesgo elevado de desarrollar DT2 del orden del 30% [7] y de 70% [9] en los siguientes 4 y 30 años, respectivamente.

El desarrollo de la DT2 es un proceso lento y progresivo condicionado por factores genéticos, ambientales y de comportamiento. Actualmente no existe una cura definitiva para esta enfermedad. Sin embargo, varios estudios han demostrado que se puede prevenir o demorar su aparición en personas con PDM a través de la adopción de un estilo de vida saludable (plan de alimentación y práctica regular de actividad física) y/o asociado con la ingesta de diversos fármacos [7,26]. En Argentina, una iniciativa de este tipo es el Programa Piloto para la Prevención Primaria de Diabetes en la provincia de Buenos Aires (PPDBA) desarrollado por el CENEXA (UNLPCONICET) y financiado por el Ministerio de Ciencia y Tecnología de la Nación, la empresa SANOFI y el CONICET [12].

La detección de DT2 y PDM representa un verdadero desafío para la medicina debido a la ausencia de síntomas patogenómicos y a la falta de conocimiento de los factores de riesgo asociados. Es por eso que frecuentemente una persona pueda pasar meses (o incluso años) sin saber que se encuentra en riesgo. En ese sentido, estadísticas publicadas en el año 2018 por la Federación Internacional de Diabetes muestran que aproximadamente un 50% de la población mundial desconoce su enfermedad [10]. Esto explica la necesidad de contar con un método de detección simple y preciso. En consecuencia, este artículo propone desarrollar y evaluar modelos predictivos basados en aprendizaje automático (AA) que permitan identificar personas con riesgo de diabetes y PDM en la población argentina considerando como base de datos la correspondiente al PPDBA.

El resto del artículo se organiza de la siguiente forma. La Sección 2 introduce el marco referencial para este trabajo. Luego, la Sección 3 describe el procesamiento realizado a la base de datos mientras que la Sección 4 describe y analiza los resultados obtenidos. Finalmente, la Sección 5 presenta las conclusiones junto al trabajo futuro.

2. Marco Referencial

2.1 Factores de Riesgo y Diagnóstico de Diabetes y Prediabetes

El desarrollo de DT2 es un proceso lento y progresivo que se encuentra condicionado por factores genéticos, ambientales y de comportamiento. Entre los factores de riesgo se encuentran, entre otros, el género, el índice de masa corporal (IMC), la circunferencia de cintura, los hábitos de alimentación, la práctica de actividad física, la edad, los antecedentes familiares de diabetes (incluida gestacional), la etnia y los trastornos del sueño. Por su parte, la PDM representa un estado previo a la DT2 y su progresión puede prevenirse e incluso revertirse mediante la adopción de estilos de vida saludables [27]. Si además consideramos que la PDM no es una pre-enfermedad, pues ya presenta disfunciones metabólicas, resulta sumamente importante identificar personas con PDM tanto como lo es poder hacerlo con las que ya tienen DT2 no diagnosticada.

El diagnóstico se realiza por medio de análisis de sangre y cualquier persona con presencia de los síntomas o factores de riesgo asociados debe ser examinada. En Argentina, habitualmente se emplea la Prueba de Tolerancia Oral a la Glucosa (PTOG) para determinar si una persona posee DT2, PDM o ninguna de ellas [1].

2.2 Trabajos relacionados

En la última década, numerosos modelos han sido propuestos para identificar diabetes no diagnosticada y/o PDM utilizando técnicas de AA. Estas propuestas utilizan diversas variables clínicas y de laboratorio asociadas a los factores de riesgo de la enfermedad para establecer la predicción. La mayoría de ellas [15] [2] [13] [16] [19] [21] [16] [33] [6] [28] [22] [24] [18] [34] [30] [25] emplean una base de datos de diabetes conocida como PIMA Indian Diabetes (PID) del repositorio de la Universidad de California Irvine [2], EEUU. Este *dataset* contiene registros de mujeres del pueblo indígena Pima de EEUU y se compone de 8 atributos relacionados con los factores de riesgo de desarrollar la enfermedad. Dispone de un total de 786 registros; sin embargo, el número se reduce a la mitad al eliminar aquellos que poseen valores nulos. Se puede decir que estos trabajos son una *prueba de concepto* más que una implementación real y que, en general, se han orientado a mejorar el rendimiento (*accuracy*) de los algoritmos de clasificación de diabetes.

Son pocos los trabajos que no emplean la base de datos PID. Entre ellos se encuentran [14] [32] [8] quienes propusieron modelos para identificar diabetes y prediabetes usando datos de la Encuesta NHANES de EEUU. Recientemente, [31] también estudió la aplicación de distintas técnicas de AA para la detección de diabetes no diagnosticada en la población estadounidense aunque empleando la base de datos BRFSS. En forma similar, [20] presentó diferentes modelos predictivos de estas enfermedades para la población china usando una base de datos *ad-hoc*. A diferencia de los anteriores, [5] hizo hincapié en el desarrollo y validación de modelos predictivos únicamente para prediabetes.

Como se explicó en la Sección 2.1, el desarrollo de DT2 se encuentra condicionado por factores que pueden variar de una población a otra. Es por lo que un modelo predictivo que resulte adecuado para una población, no necesariamente lo será para otra. Este estudio representa el primer paso hacia modelos predictivos específicos para la población argentina.

3. Implementación

3.1 Conjunto de datos

[1] Esta prueba comienza con una extracción de sangre de la persona en ayunas. Luego, se le pedirá que tome un líquido que contiene una cierta cantidad de glucosa. A continuación, se le tomarán muestras de sangre nuevamente cada 30 a 60 minutos después de ingerir la solución. El examen es costoso económicamente y puede demorar hasta 3 horas.

[2] https://archive.ics.uci.edu/ml/index.php

Descripción Los modelos predictivos serán desarrollados a partir de la base de datos del programa PPDBA [12], la cual cuenta con 1316 registros de personas. Cada registro corresponde a una persona que mediante PTOG fue identificada como diabética, prediabética o sin ninguna de ellas. Además de datos de laboratorio (hemoglobina glicosilada; colesterol total, HDL y LDL; triglicéridos y creatinina), se cuenta con variables clínicas asociadas a los factores de riesgo de estas enfermedades tales como el sexo; la edad; el Indice de Masa Corporal (IMC); la presión arterial; los antecedentes familiares de diabetes; los hábitos alimenticios y de actividad física; entre otros.

Caracterización De los 1316 registros actuales, 80 debieron ser descartados ya que omitían valores requeridos de glucemia para poder calcular el resultado de la PTOG (no es posible determinar la clase). La Tabla 1 presenta una breve descripción estadística de los 1236 registros disponibles. Se puede notar que hay varias variables que presentan nulos, los cuales se analizan con mayor profundidad en la sección siguiente. También se puede observar que: hay más personas del sexo femenino que del masculino; la mayoría de las personas tienen entre 45-64 años; la mayoría de las personas tienen un IMC mayor a 30 kg/m^2; la mayoría de las personas tienen una circunferencia de cintura de más de 102cm y de 88cm para el sexo masculino y femenino, respectivamente; la mayoría de las personas realizan actividad física; consumen vegetales, frutas y hortalizas; no toman medicación para controlar hipertensión; sí le encontraron hiperglucemia en algún control; y alguno de sus familiares (de primer o segundo grado) tiene diabetes; las variables asociadas a glucemia basal, colesterol HDL, triglicéridos y creatinina basal parecieran tener una amplia dispersión; el resto no; en cuanto a la clase, la mitad de las personas no están en riesgo de tener prediabetes o diabetes.

Limpieza

Valores con ruido Para analizar la presencia de ruido de las variables, se utilizó el método de Tukey para identificar los intervalos de valores atípicos leves y extremos. En base a lo anterior, se detectaron valores atípicos leves en las variables edad (4), imc (10), circ___de_cintura (1), glucemia_basal (27), glucemia_pprandial (46), colesterol_ldl (7), colesterol_total (9), colesterol_hdl (11), triglicéridos (21) y creatinina_basal (8). Además, se encontraron valores atípicos extremos en glucemia_basal (51), glucemia_pprandial (6), trigliceridos (12) y creat_basal (6).

Valores nulos En la Tabla 1 se puede observar que no hay valores nulos en las variables cualitativas, a excepción de le_diag___familiar, que presenta 3 registros con valores nulos. No ocurre lo mismo con las variables cuantitativas, donde el faltante de valores es mucho mayor. Afortunadamente, para algunas de esas variables cuantitativas, sí se cuenta con una variable cualitativa asociada que permite conocer el rango en que se encuentra ese valor faltante. Esto ocurre concretamente en los casos de edad, imc y circ__de_cintura. El resto de las variables que presentan nulos son: glucemia_pprandial, colesterol_total, colesterol_ldl, colesterol_hdl, trigliceridos, creat_basal y hem___glicosilada.

Transformaciones

Tabla 1. Breve descripción estadística del dataset

Variable	# nulos	Medida	# casos
sexo	0	Masculino (%)	395 (32%)
		Femenino (%)	841 (68%)
edad	564	Media+DE	57.23±8.8428
rango_edad	0	Menos de 45 años (%)	205 (17%)
		45-54 años (%)	435 (35%)
		54-64 años (%)	429 (34%)
		Mayor de 64 años (%)	167 (14%)
imc	619	Media+DE	31.65±6.314
rango_imc	0	Menor de 25kg/m2	93 (8%)
		25-30 kg/m2	319 (26%)
		Mayor de 30 kg/m2	824 (66%9
circ___cintura	1181	Media+DE	101.3091±13.55
rango___cintura	0	Menos de 94/80 cm (M/F) (%)	56 (4%)
		M: 94-102cm / F: 80-88cm (%)	205 (17%)
		M: Más de 102/88cm (M/F) (%)	975 (79%)
actividad_fisica	0	Sí (%)	915 (74%)
		No (%)	321 (26%)
cons___hortalizas	0	Sí (%)	821 (66%)
		No (%)	415 (34%)
toma___hta	0	Sí (%)	497 (40%)
		No (%)	739 (60%)
le___hiperglucemia	0	Sí (%)	999 (81%)
		No (%)	237 (19%)
le_diag___familiar	3	No (%)	395 (32%)
		Primer grado (%)	412 (33%)
		Segundo grado (%)	426 (34%)
glucemia_basal	0	Media+DE	104.36±27.28
glucemia_pprandial	55	Media+DE	119,59±42,51
colesterol_total	705	Media+DE	198,28±41,1
colesterol_ldl	715	Media+DE	119,79±36,82
colesterol_hdl	706	Media+DE	49,82±14,40
trigliceridos	705	Media+DE	151,4±95,59
creatinina_basal	619	Media+DE	1,117±5,8
hem___glucosilada	635	Media+DE	5.61±0,43
resultado_ptog	0	Sin riesgo (%)	620 (50%)
		Prediabetes (%)	480 (38%)
		Diabetes (%)	136 (12%)

Tratamiento de valores atípicos Se consultó con expertos del dominio médico sobre las ocurrencias de valores atípicos en las determinaciones de laboratorio. Se concluyó que, si bien son valores que estadísticamente pueden ser considerados atípicos, sí se

encuentran dentro de los posibles valores extremos para estas determinaciones. La excepción son 2 valores en creatinina_basal (’85’, ’118’) que efectivamente corresponden a (posibles) errores de carga. Por lo tanto, esos dos valores particulares se reemplazaron con nulos.

Tratamiento de valores nulos Se debe optar entre: (1) eliminar las variables que tienen nulos; o (2) eliminar los registros que tienen variables con valores nulos. La opción 1 permite mantener el tamaño muestral a costa de reducir la cantidad de variables de entrada para los modelos. En sentido opuesto, la opción 2 permite mantener la cantidad de características a costo de reducir el tamaño muestral. En este caso, se decidió seleccionar la opción 2 únicamente para la variable le_diag___familiar, que cuenta con sólo 3 valores nulos. Por otro lado, la opción 1 se aplicó a las variables edad, imc y circ_de_cintura, considerando que hay una variable cualitativa asociada que permite conocer el rango de cada valor. Para el resto de las variables, se discutirá en la Sección 3.2.

Agrupamiento de variable de clase De la Tabla 1 se puede notar que la distribución de clases no está balanceada (variable resultado_ptog). Para minimizar el impacto de esta cuestión, se procede a crear una nueva variable de clase que divida entre personas sin riesgo de tener PDM o DM (registros con valor “Normal”) y las que sí lo tienen (registros con valor “PDM” o “DM”). Como resultado, la variable queda balanceada en cuanto a ocurrencias de cada valor y, a la vez, se simplifica el análisis posterior al pasar a ser ahora un problema de clasificación binaria. Sin embargo, se deberá ser cuidadoso al momento de analizar los resultados, especialmente con lo que se pueda decir sobre predicción de DM, por ser la de menor ocurrencia.

Eliminación de variables resultado_ptog fue descartada por el agrupamiento realizado en la sección anterior, mientras que glucemia_pprandial fue excluida, ya que contar con ese valor implicaría que la persona tuviera que hacerse una PTOG, careciendo de sentido el uso de los modelos propuestos.

Análisis de correlaciones La Fig. 1 muestra la matriz de correlación obtenida sobre el *dataset* inicial disponible. Desde el punto de vista clínico, tiene sentido que exista una correlación lineal débil entre el rango de la circunferencia de cintura y el rango IMC. También resulta razonable que existan correlaciones débiles entre los rangos de edad, IMC y circunferencia de cintura y sus valores asociados en las variables de edad, imc y circ___de_cintura. Por otra parte, el colesterol_total se calcula a partir del colesterol_ldl [3], lo que explica su correlación fuerte. Adicionalmente, la relación entre glucemia_basal, glucemia_pprandial, resultado_ptog y clase tiene sentido, ya que las dos primeras determinan el valor de la tercera, la cual a su vez se agrupa para generar la cuarta de ellas.

3.2 Segmentaciones propuestas

Ante el porcentaje alto de nulidad de las variables de laboratorio, se planteó la posibilidad de generar varios datasets a partir del original, considerando distintos criterios para el tratamiento de registros nulos:

- Dataset Clínica+Laboratorio (DCL-bin). Conjunto de datos al cual se realizó una eliminación de registros completos, derivando en 16 variables con 503 ejemplos (229 ⇒ Sin riesgo, 274 ⇒ Con riesgo). Este dataset mantiene todos las variables

disponible (datos clínicos y de laboratorio) a costa de perder cantidad de registros.

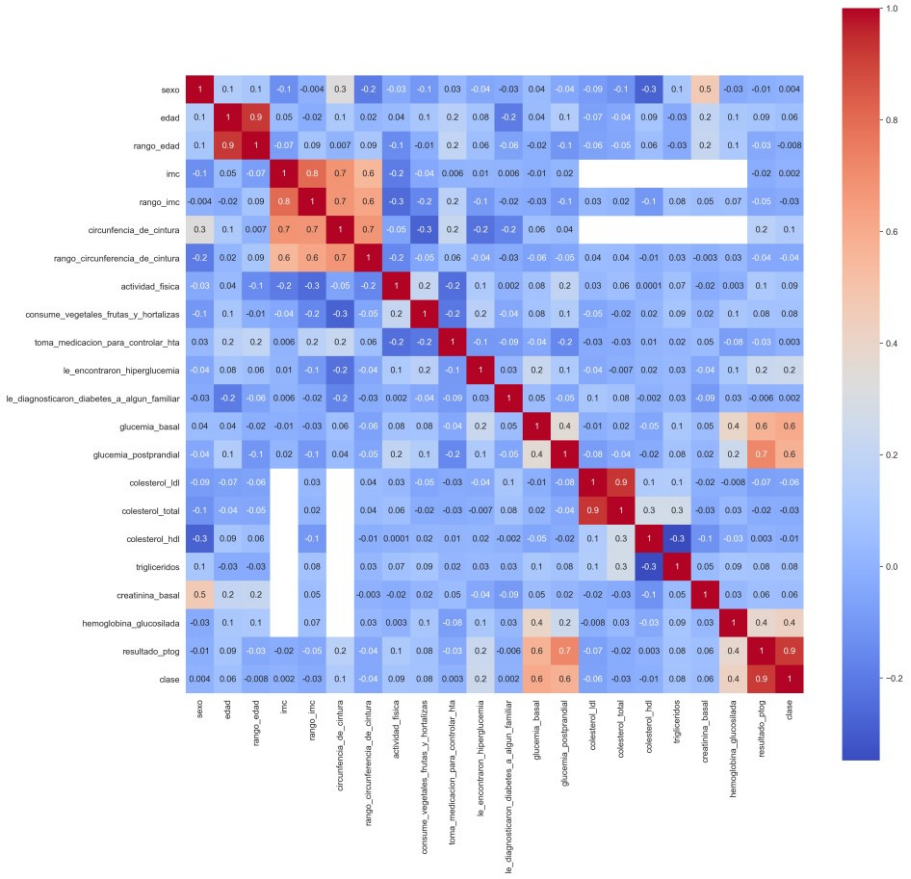


Figura1. Matriz de correlación sobre dataset inicial

- Dataset Clínica+Glucemia basal (DCG-bin). Conjunto de datos que mantiene la información clínica disponible y la única variable de laboratorio que no posee valores nulos (glucemia_basal); el resto de las ellas fueron eliminadas. De esta forma, este dataset cuenta con 10 variables y 1233 ejemplos. En contraposición con DCL aquí se preserva la cantidad de registros frente al valor aportado por el resto de los variables de laboratorio.

4. Resultados Experimentales

4.1 Diseño experimental

Se llevó a cabo un proceso de exploración, preprocesamiento, entrenamiento y evaluación de diversos algoritmos utilizando *scikit-learn*. En particular, se consideraron Logistic Regression (LR), Decision Tree (DT) con max_depth = 5 y class_weight = balanced, k-Nearest Neighbor (kNN) con n_neighbors = 7 y Random Forest (RF) con max_depth = 2. Asimismo, se utilizó TensorFlow para la construcción de modelos de Artificial Neural Networks (ANN) con epochs = 60 y batch_size = 16. En particular, se construyó una ANN de una capa oculta con 100 neuronas y función de activación ReLU, con una regularización L2 de 0,1 para controlar el sobreajuste. Se utilizó un optimizador Adamax con una tasa de aprendizaje de 0,001.

Siguiendo lineamientos habituales del área, el conjunto de datos fue particionado en dos: entrenamiento (70%) y evaluación (30%). Para reducir posibles sesgos, se aplicó la técnica de validación cruzada con muestreo estratificado aleatorio (*StratifiedShuffleSplit, n_splits=50* [23]). Debido a que las variables de entrada poseen magnitudes diferentes, se aplicó una normalización *min-max* para todos los modelos excepto para la ANN donde se utilizó una normalización estándar.

Todas las experimentaciones se realizaron sobre una computadora local utilizando *Jupyter Notebook* [17]. El hardware disponible fue una CPU Intel Core i7 2.6 GHz Quad-Core, 16 GB de memoria RAM y sistema operativo macOS.

4.2 Modelos y resultados para DCL-bin

Tabla 2. Resultados (evaluación) para los modelos aplicados a DCL-bin (clase positiva = "Con riesgo")

Modelo	Accuracy	Precision	Recall	F-score	AUC
RF	94.58 ± 1.51	98.87 ± 1.3	91.1 ± 2.93	94.79 ± 1.53	0.95
DT	93.42 ± 1.59	96.02 ± 2.02	91.73 ± 2.65	93.79 ± 1.53	0.94
ANN	91.13 ± 1.98	92.29 ± 2.68	91.39 ± 2.77	91.79 ± 1.84	0.91
LR	85.56 ± 3.21	90.5 ± 3.11	82.1 ± 4.81	86.01 ± 3.3	0.93
kNN	71.62 ± 3.82	75.53 ± 3.87	70.83 ± 5.75	72.97 ± 4	0.72

La Tabla 2 muestra las métricas de rendimiento para los diferentes modelos aplicados al dataset DCL-bin considerando como clase positiva a "Con riesgo". De los 5 modelos, se puede observar que hay 3 opciones que obtienen valores de *accuracy* superiores al 90%, lo que significa que aproximadamente 9 de cada 10 de los registros evaluados del total, fueron clasificados correctamente (los desvíos son menores al 2%). De estas 3 opciones, RF es el que obtiene la mejor *accuracy*, seguido de DT y finalmente ANN. Mientras que el rendimiento de LR se encuentra cercano a los anteriores con 86% de *accuracy*, el correspondiente a kNN es pobre al alcanzar sólo 72%.

En cuanto a *precision*, los valores se condicen con los de *accuracy*. RF, DT y ANN presentan valores superiores al 90% para la clase de interés ("Con riesgo") lo que significa que, con cualquiera de las opciones, a más de 9 personas a las que se les dice que tiene riesgo, realmente lo tiene. En el mismo sentido, los 3 modelos mencionados valores de *recall* cercanos al 91%, lo que significa que sólo 1 de cada 10 personas que tiene riesgo, no es identificada por ellos.

Por último, al momento de seleccionar un modelo determinado, puede ser interesante examinar los valores de AUC además de la *accuracy*. En este caso, se puede notar que RL desplaza a ANN como uno de los 3 que obtiene los valores más altos de AUC, en comparación a los de mejor *accuracy*.

4.3 Modelos y resultados para DCG-Bin

Tabla 3. Resultados (evaluación) para los modelos aplicados a DCG-bin (clase positiva = "Con riesgo")

Modelo	Accuracy	Precision	Recall	F-score	AUC
RF	93.23 ± 1.12	100 ± 0	86.38 ± 2.25	92.68 ± 1.3	0.93
DT	91.88 ± 1.67	96.68 ± 2.74	86.72 ± 2.2	91.39 ± 1.73	0.92
ANN	91.04 ± 1.63	94.45 ± 2.47	87.17 ± 2.31	90.64 ± 1.71	0.91
LR	75.66 ± 4.37	80.96 ± 4.44	66.7 ± 6.55	73.05 ± 5.3	0.83
kNN	69.97 ± 2.01	70.63 ± 2.32	67.92 ± 3.29	69.2 ± 2.23	0.7

La Tabla 3 muestra las métricas de rendimiento para los diferentes modelos aplicados al dataset DCG-bin considerando como clase positiva a "Con riesgo". Al igual que con DCL-bin, RF, DT y ANN obtienen muy buenos valores de *accuracy*, siendo superiores al 90% (los desvíos son menores al 2%). kNN vuelve a presentar un rendimiento pobre con 70% de *accuracy*, mientras que LR se encuentra entre los 3 anteriores y kNN, aunque con un rendimiento más bajo que con el dataset anterior. En cuanto a *precision*, RF sobreajusta al obtener 100% para la clase "Con riesgo". Por su parte, DT y ANN presentan valores superiores al 90% para la misma clase, estando LR y kNN bastante debajo de ese valor. Al analizar los valores de *recall*, tanto RF como DT y ANN son los que consiguen los mejores valores, estando cercanos al 90% para la clase de interés.

Por último, y a diferencia de DCL-bin, los 3 que obtienen los valores más altos de AUC coinciden con los de mejor *accuracy*.

4.4 Discusión

Algunos de los modelos desarrollados obtuvieron muy buenos rendimientos para ambos datasets. En particular, RF, DT y ANN demostraron gran poder de clasificación, con altos valores en las métricas consideradas. En ese sentido, resulta importante aclarar que los modelos propuestos no pretenden reemplazar a las PTOGs como mecanismo de diagnóstico de DT2 y PDM. Al ser enfermedades de difícil detección precoz, estos modelos buscan identificar aquellas personas de la población argentina que tengan alta probabilidad de tenerlas y desconozcan su condición. Para confirmar el diagnóstico, las personas identificadas deberán realizar eventualmente una PTOG. Los modelos ayudarían a identificar a quienes deben realizarlo y suplirían la ausencia de herramientas de este tipo.

Se puede notar que no hay diferencias significativas entre los mejores valores de *accuracy* y *F-score* conseguidos para ambos datasets. Aunque no es (del todo) correcto comparar resultados de modelos entrenados con datasets diferentes, esta cuestión

podría tener incidencia en el costo de llevar a la práctica los modelos, considerando que conseguir las variables de laboratorio no es gratuito ni sencillo. Para poder dilucidarla, sería necesario contar con un mayor número de registros sin nulos.

Por último, una cuestión importante que debe tenerse en cuenta es el agrupamiento de PDM y DM como clase única, lo cual favoreció al balanceo y simplificó el problema al volverlo de clasificación binaria. El costo es justamente no poder diferenciar entre los casos de PDM y DM. Sin embargo, desde un punto de vista médico, esto no sería tan grave, ya que a fin de cuentas lo que interesa es identificar quienes están en riesgo (no importa de cuál).

5. Conclusiones y Trabajo Futuro

Considerando que DM y PDM son enfermedades de difícil detección, en este trabajo se desarrollaron y evaluaron modelos predictivos específicos para la población argentina a partir de la base de datos del PPDBA. En primer lugar, fue necesario realizar un cuidadoso preprocesamiento de la base de datos, lo que derivó en la generación de dos datasets particulares (DCL-bin y DCG-bin) considerando el compromiso entre cantidad de variables y de registros disponibles. Luego, se aplicaron 5 modelos de clasificación diferentes a cada uno de ellos. Los resultados obtenidos muestran que algunos de los modelos propuestos obtuvieron muy buenos rendimientos para ambos datasets. En particular, RF, DT y ANN demostraron gran poder de clasificación, con altos valores en las métricas consideradas. Debido a limitaciones propias de la base de datos, no es posible afirmar que los resultados sean concluyentes, aunque sí resultan promisorios. Considerando la vacancia de herramientas de este tipo para la población argentina, este trabajo representa el primer paso hacia modelos más sofisticados.

Entre las líneas de trabajo futuro se encuentran:

- Conseguir más registros de la base de datos para mejorar su calidad y al mismo tiempo aumentar su representatividad, para luego replicar el estudio realizado.
- Evaluar el rendimiento de modelos generados a partir de una nueva segmentación que sólo considere datos clínicos. Un modelo de estas características sería más sencillo, sin costo y factible de realizar en cualquier momento, aunque probablemente de menor rendimiento.
- Considerar el desarrollo de modelos de clasificación multiclase para separar los casos de DM y PDM.

Financiamiento. Este estudio fue parcialmente respaldado por PICT-2020-SERIE-A00901.

Referencias

1. Professional practice committee: Standards of medical care in diabetes—2021. Diabetes Care 44 (Supplement 1), S3–S3 (2021). https://doi.org/10.2337/dc21-Sppc, https://care.diabetesjournals.org/content/44/Supplement1/S3

2. Al Jarullah, A.A.: Decision tree discovery for the diagnosis of type ii diabetes. In: 2011 International Conference on Innovations in Information Technology. pp. 303–307 (April 2011). https://doi.org/10.1109/INNOVATIONS.2011.5893838
3. Association, A.H.: What Your Cholesterol Levels Mean. https://www.heart.org/en/healthtopics/cholesterol/about-cholesterol/what-your-cholesterol-levels-mean (2020), accedido: 2022-10-10
4. Bolin, K., Gip, C., Mörk, A.C., Lindgren, B.: Diabetes, healthcare cost and loss of productivity in sweden 1987 and 2005 - a register-based approach. Diabetic medicine : a journal of the British Diabetic Association 26, 928–34 (10 2009). https://doi.org/10.1111/j.14645491.2009.02786.x
5. Choi, S.B., Kim, W.J., Yoo, T.K., Park, J.S., Chung, J.W., Lee, Y.h., Kang, E.S., Kim, D.W.: Screening for prediabetes using machine learning models. Computational and mathematical methods in medicine 2014 (2014)
6. Dey, S.K., Hossain, A., Rahman, M.M.: Implementation of a web application to predict diabetes disease: An approach using machine learning algorithm. In: 2018 21st International Conference of Computer and Information Technology (ICCIT). pp. 1–5 (2018). https://doi.org/10.1109/ICCITECHN.2018.8631968
7. Diabetes Prevention Program Research Group: Reduction in the incidence of type 2 diabetes with lifestyle intervention or metformin. New England Journal of Medicine 346(6), 393–403 (2002). https://doi.org/10.1056/NEJMoa012512
8. Dinh, A., Miertschin, S., Young, A., Mohanty, S.D.: A data-driven approach to predicting diabetes and cardiovascular disease with machine learning. BMC Medical Informatics and Decision Making 19(1), 211 (Nov 2019). https://doi.org/10.1186/s12911-019-0918-5
9. Eddy, D.M., Schlessinger, L., Kahn, R.: Clinical Outcomes and Cost-Effectiveness of Strategies for Managing People at High Risk for Diabetes. Annals of Internal Medicine 143(4), 251–264 (08 2005). https://doi.org/10.7326/0003-4819-143-4-200508160-00006
10. Federación Internacional de Diabetes: Atlas de la diabetes de la fid. Tech. rep., Federación Internacional de Diabetes (2017), https://diabetesatlas.org/resources/2017-atlas.html
11. Gagliardino, J.J., Assad, D., Gagliardino, G.G., Kronsbein, P., Lahera, E., Mercuri, N., Rizzuti, L., Zufriategui, Z.: Cómo tratar mi diabetes. Buenos Aires, Argentina, 3 edn. (11 2016)
12. Gagliardino, J.J., Etchegoyen, G., Bourgeois, M., Fantuzzi, G., García, S., González, L., Elgart, J.F., Ré, M., Ricart, A., Ricart, J.P., Spinedi, E.: Prevención primaria de diabetes tipo 2 en argentina: estudio piloto en la provincia de buenos aires. Revista Argentina de Endocrinología y Metabolismo 53(4), 135 – 141 (2016). https://doi.org/https://doi.org/10.1016/j.raem.2016.11.002
13. Hashi, E.K., Zaman, M.S.U., Hasan, M.R.: An expert clinical decision support system to predict disease using classification techniques. In: 2017 International Conference on Electrical, Computer and Communication Engineering (ECCE). pp. 396–400 (Feb 2017). https://doi.org/10.1109/ECACE.2017.7912937
14. Heikes, K.E., Eddy, D.M., Arondekar, B., Schlessinger, L.: Diabetes risk calculator. Diabetes Care 31(5), 1040–1045 (2008). https://doi.org/10.2337/dc07-1150
15. Ilango, B.S., Ramaraj, N.: A hybrid prediction model with f-score feature selection for type ii diabetes databases. In: Proceedings of the 1st Amrita ACM-W Celebration on Women in Computing in India. pp. 13:1–13:4. ACM, New York, NY, USA (2010). https://doi.org/10.1145/1858378.1858391
16. Jayanthi, N., Babu, B.V., Rao, N.S.: Survey on clinical prediction models for diabetes prediction. Journal of Big Data 4(1), 26 (2017). https://doi.org/10.1186/s40537-017-00827
17. Jupyter Notebook Documentation: 7.0.0rc2 documentation. https://jupyternotebook.readthedocs.io/en/latest/index.html, accedido: 2023-04-18

18. Kaur, H., Kumari, V.: Predictive modelling and analytics for diabetes using a machine learning approach. Applied Computing and Informatics (2018). https://doi.org/https://doi.org/10.1016/j.aci.2018.12.004
19. Maniruzzaman, M., Kumar, N., Menhazul Abedin, M., Shaykhul Islam, M., Suri, H.S., El-Baz, A.S., Suri, J.S.: Comparative approaches for classification of diabetes mellitus data: Machine learning paradigm. CMPB 152, 23–34 (2017). https://doi.org/https://doi.org/10.1016/j.cmpb.2017.09.004
20. Meng, X.H., Huang, Y.X., Rao, D.P., Zhang, Q., Liu, Q.: Comparison of three data mining models for predicting diabetes or prediabetes by risk factors. The Kaohsiung Journal of Medical Sciences 29(2), 93 – 99 (2013). https://doi.org/https://doi.org/10.1016/j.kjms.2012.08.016
21. Mercaldo, F., Nardone, V., Santone, A.: Diabetes mellitus affected patients classification and diagnosis through machine learning techniques. Procedia Computer Science 112, 2519–2528 (2017). https://doi.org/https://doi.org/10.1016/j.procs.2017.08.193
22. Mir, A., Dhage, S.N.: Diabetes disease prediction using machine learning on big data of healthcare. In: 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA). pp. 1–6 (2018). https://doi.org/10.1109/ICCUBEA.2018.8697439
23. Scikit-Learn: 3.1. Cross-validation: evaluating estimator performance. https://scikitlearn.org/stable/modules/crossvalidation.html,accedido : 2023 − 04 − 11
24. Sisodia, D., Sisodia, D.S.: Prediction of diabetes using classification algorithms. Procedia Computer Science 132, 1578 – 1585 (2018). https://doi.org/https://doi.org/10.1016/j.procs.2018.05.122
25. Sneha, N., Gangil, T.: Analysis of diabetes mellitus for early prediction using optimal features selection. Journal of Big Data 6(1), 13 (2019). https://doi.org/10.1186/s40537019-0175-6
26. Tuomilehto, J., Lindström, J., Eriksson, J.G., Valle, T.T., Hämäläinen, H., Ilanne-Parikka, P., Keinänen-Kiukaanniemi, S., Laakso, M., Louheranta, A., Rastas, M., Salminen, V., Aunola, S., Cepaitis, Z., Moltchanov, V., Hakumäki, M., Mannelin, M., Martikkala, V., Sundvall, J., Uusitupa, M.: Prevention of type 2 diabetes mellitus by changes in lifestyle among subjects with impaired glucose tolerance. New England Journal of Medicine 344(18), 1343–1350 (2001). https://doi.org/10.1056/NEJM200105033441801
27. Vistisen, D., Kivimäki, M., Perreault, L., Hulman, A., Witte, D.R., Brunner, E.J., Tabák, A., Jørgensen, M.E., Færch, K.: Reversion from prediabetes to normoglycaemia and risk of cardiovascular disease and mortality: the whitehall ii cohort study. Diabetologia 62(8), 1385–1390 (Aug 2019). https://doi.org/10.1007/s00125-019-4895-0
28. Wei, S., Zhao, X., Miao, C.: A comprehensive exploration to the machine learning techniques for diabetes identification. In: 2018 IEEE 4th World Forum on Internet of Things (WF-IoT). pp. 291–295 (2018). https://doi.org/10.1109/WF-IoT.2018.8355130
29. Williams, R., Van Gaal, L., Lucioni, C.: Assessing the impact of complications on the costs of type ii diabetes. Diabetologia 45(1), S13–S17 (Jul 2002). https://doi.org/10.1007/s00125-002-0859-9
30. Wu, H., Yang, S., Huang, Z., He, J., Wang, X.: Type 2 diabetes mellitus prediction model based on data mining. Informatics in Medicine Unlocked 10, 100 – 107 (2018). https://doi.org/https://doi.org/10.1016/j.imu.2017.12.006
31. Xie, Z., Nikolayeva, O., Luo, J., Li, D.: Building risk prediction models for type 2 diabetes using machine learning techniques. Preventing chronic disease 16, E130–E130 (Sep 2019). https://doi.org/10.5888/pcd16.190109
32. Yu, W., Liu, T., Valdez, R., Gwinn, M., Khoury, M.J.: Application of support vector machine modeling for prediction of common diseases: the case of diabetes and pre-diabetes. BMC Medical Informatics and Decision Making 10(1), 16 (2010). https://doi.org/10.1186/14726947-10-16

33. Yuvaraj, N., SriPreethaa, K.R.: Diabetes prediction in healthcare systems using machine learning algorithms on hadoop cluster. Cluster Computing 22(1), 1–9 (Jan 2019). https://doi.org/10.1007/s10586-017-1532-x
34. Zou, Q., Qu, K., Luo, Y., Yin, D., Ju, Y., Tang, H.: Predicting diabetes mellitus with machine learning techniques. Frontiers in genetics 9, 515–515 (Nov 2018). https://doi.org/10.3389/fgene.2018.00515

Coordinadores:

- María Luján Ganuza (UNS)
- Roberto Guerrero (UNSL)
- Oscar Bría (INVAP)

CACiC 2023

CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

XXI Workshop de Computación Gráfica, Imágenes y Visualización

> Full Papers

WCGIV

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Visual Analysis of Temporal Data Associated with Cryptocurrencies

Mercedes Barrionuevo[1] and María Luján Ganuza[2]

[1] Universidad Nacional de San Luis,
mdbarrio@unsl.edu.ar
[2] Inst. for Computer Science and Engineering, ICIC (CONICET-UNS),
San Andrés 800, 8000 Bahía Blanca, Argentina,
mlg@cs.uns.edu.ar

Abstract. Cryptocurrencies have emerged as a novel and dynamic asset class in the financial landscape, attracting significant attention from investors and researchers alike. As the popularity and diversity of cryptocurrencies continue to grow, there is an increasing need to analyze and understand the temporal dynamics and patterns within their data. Therefore, the treatment of such data involves the application of a wide range of tools for analysis and exploration, including different visualization techniques. In this paper, we present CryptoVisualizer, an integral visual analysis tool for temporal data associated with cryptocurrencies. Our proposal integrates different visualization methods and interactions to empower users in exploring and understanding temporal patterns, trends, and relationships in cryptocurrency markets.

Keywords: Visualization. Temporal Data Visualization. Cryptocurrencies. Visual Data Analysis.

1. Introduction

Visualization of temporal data refers to the process of using visual representations to explore and understand data that varies over time. It involves examining patterns, trends, and relationships within the temporal data to gain insights and extract meaningful information. In various academic disciplines understanding temporal patterns and trends is essential for making informed decisions and extracting significant insights. By using visualizations, researchers can examine and compare data at different time scales, facilitating the interpretation and analysis of information [1,2].

The visualization of temporal data also contributes to effectively communicating scientific findings. Temporal visualizations can help illustrate significant changes, demonstrate the evolution of variables over time, and identify trends and patterns that occur over time. The visualization of temporal data has emerged as a crucial area of research and development in recent years, driven by the increasing availability of time-series data across various domains [3]. Several traditional visualization techniques have been applied to this kind of data,

(like line charts, bar charts, and heat maps, among others) and dedicated techniques have been presented [4,5,6]. In this context, Andrienko and Andrienko [7] proposed a taxonomy of tasks for time-oriented data aimed intended to help researchers and practitioners in understanding the diverse challenges and objectives of working with temporal data.

Cryptocurrencies have garnered significant attention from investors and technology experts in the field of economics. Analyzing the behavior of cryptocurrencies is of interest in the fields of economics and technology due to their impact on technological innovation, financial volatility and risk, new economic models, and their influence on economic policies. Understanding how they function and how they affect economic and financial systems is fundamental to harnessing their benefits and addressing the challenges associated with this new form of value exchange. In this context, the visual analysis of financial data has become a relevant research and development field [8]. Cryptocurrency visualization encompasses a variety of methods [9,10,8,11]. However, to the best of our knowledge, no visual analysis tools that integrate several visualization techniques and allow the integral analysis of different currencies are available in the literature.

In this context, we present *CryptoVisualizer*, a cryptocurrency visualization tool specifically focused on 4 cryptocurrencies of interest: Bitcoin (BTC), Ethereum (ETH), Cardano (ADA), and Dogecoin (DOGE).

In the next section, we define the questions that will serve as the basis for defining the *CryptoVisualizer*'s requirements. In section 3, we describe our proposal, detailing the characteristics of the data, the designed dashboards, and the supported interactions. In section 3.2, the results are presented, explaining how the developed tool addresses the questions defined in section 2. Finally, in section 4 a discussion is outlined, and in section 5 we elaborate on the conclusions and future work.

2. Requirements

After conducting a thorough analysis of the state of the art and basing on Andrienko and Andrienko's taxonomy of tasks for time-oriented data [7], we defined a set of questions that we considered essential to be answered by our tool. These questions formed the basis for the requirements of *CryptoVisualizer*.

1. **Q1:** In general terms, has the trend of the considered cryptocurrencies been constant, bullish, or bearish?
2. **Q2:** During a specific period of time, does the behavior of a cryptocurrency exhibit any relationship with other cryptocurrencies?
3. **Q3:** Is it possible to detect trends in the transaction volume for the different cryptocurrencies? Is there a relationship between the volume trends of different currencies?
4. **Q4:** Is it possible to detect periods of time during which the currencies remain stable or exhibit an upward or downward trend?

3. CryptoVisualizer

In this project we combined 4 datasets downloaded from the *Yahoo! Finance* repository [12] containing information related to the prices and transaction volumes of BTC, ETH, ADA, and DOGE between 2014 and 2023.

3.1 Availability and Design

CryptoVisualizer is a coordinated multi-view data visualization tool developed with Tableau [13] freely accessible from any browser[1]. The user interface was designed as a combination of two dashboards: the *PDash* (Prices Dashboard, see figure 1), and the *VDash* (Volumes Dashboard, see figure 2). Navigation to the *VDash* is facilitated by a dedicated button situated within the *PDash*.

The *PDash* integrates several views. Figure 1B shows two line charts representing the price trends of the closing price for the four cryptocurrencies from their inception to the present day. The difference between the two charts is the scale chosen to represent the Y-axis (the chart on the left uses a logarithmic scale, and the chart on the right uses a linear scale). An area chart represents the price growth of each cryptocurrency (see figure 1 C) Additionally, the *PDash* includes a *difference view*, a custom-designed view that allows analyzing losses and gains by representing the difference between the closing price and the opening price of each cryptocurrency over time (see figure 1D top). In the graph below, the difference in closing price compared to its value from the previous day is shown, also aiming to identify bullish or bearish price trends (see figure 1D button). The size of the rectangles encodes the magnitude of the difference, and the color indicates whether the difference is positive or negative. This dashboard also contains a filtering and selection panel (see figure 1A), that enables user interaction with the price charts and provides access to the *VDash* through a navigation button.

The *VDash* represents the average number of transactions conducted in different years, the relationship between volume and closing price, the relationship between the days of the week and the transaction volume, etc. It contains a filtering and selection panel similar to the *PDash* (see figure 2A). Figure 2B illustrates the *average volume graph*, which represents the average transaction volume per currency and per time unit using square marks, where the size and color are associated with the recorded average transaction volume. A line graph (see figure 2C) represents the variation of transaction volume over time for each currency. The *VDash* also contains a custom-designed area graph inspired by the Theme River technique [14], where the price of each currency is represented above and the transaction volume is represented below a horizontal axis (see figure 2D). Additionally, four bar charts (one for each currency) represent the transaction volume related to the days of the week (see figure 2E). Finally, a

custom-designed view (see figure 2F) represents the transaction volume of each cryptocurrency over time. Each rectangle represents the transaction volume in a specific time unit. Its size is associated with the transaction volume, and its color represents the cryptocurrency.

CryptoVisualizer offers a rich set of interactions that allow users to explore the data, such as single and multiple cryptocurrency selection, time range selection, and filtering by transaction volume range, among others.

[1] https://tinyurl.com/CryptoVisualizer (Spanish version)

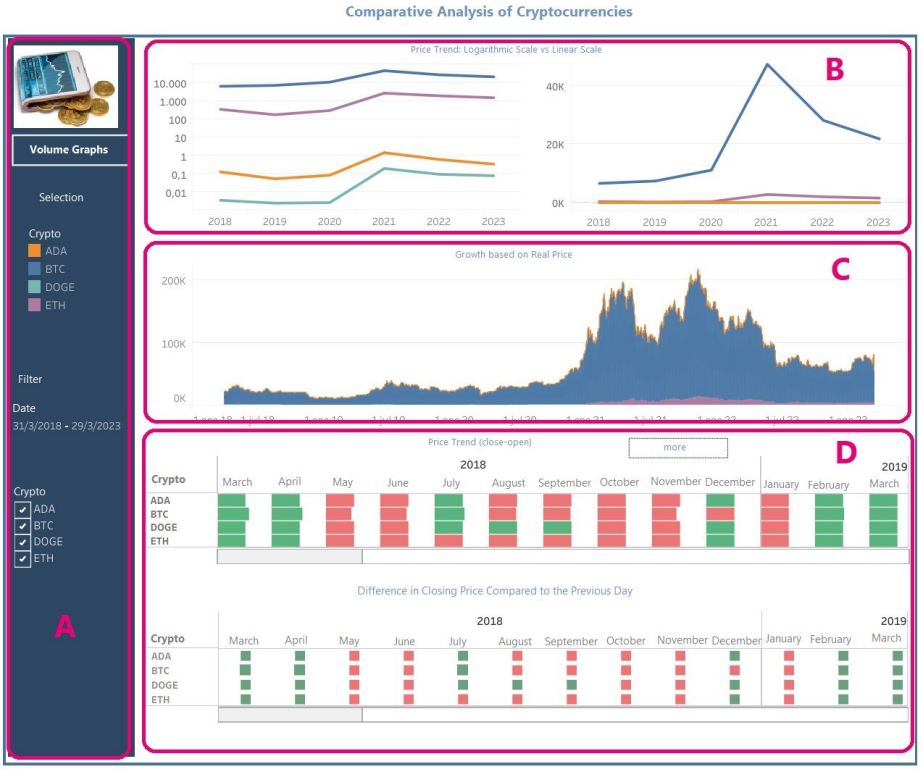


Fig.1. The Prices Dashboard (*PDash*)

3.2 Results

For price trend analysis **(Q1)**, line and area charts corresponding to the *PDash* were used (figure 3A). This view displays the behavior of the closing price of the four cryptocurrencies simultaneously, where, in general terms, from 2020 onwards, overall bullish growth is observed. However, for a more detailed analysis, filtering has been applied for each cryptocurrency (see figures 3B-E for BTC, ETH, ADA, and DOGE respectively).

For behavioral patterns detection **(Q2)**, the *difference view* in the *PDash* was used. Figure 4 illustrates an example where the same average annual behavior of the four cryptocurrencies can be observed in 2017, 2018, 2020, 2021, and 2022.

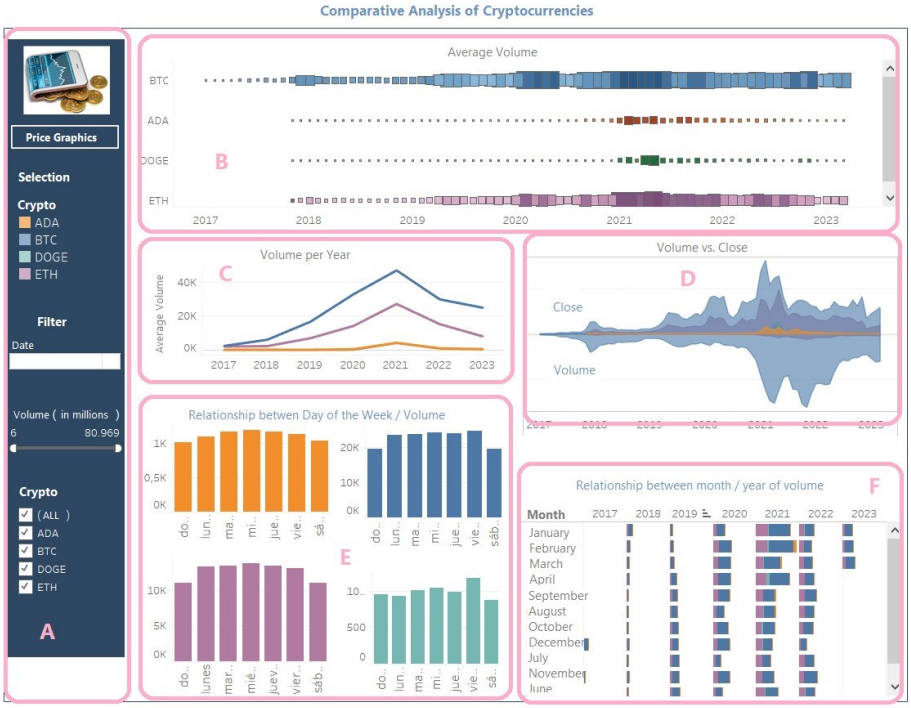


Fig.2. Volumes Dashboard (*VDash*)

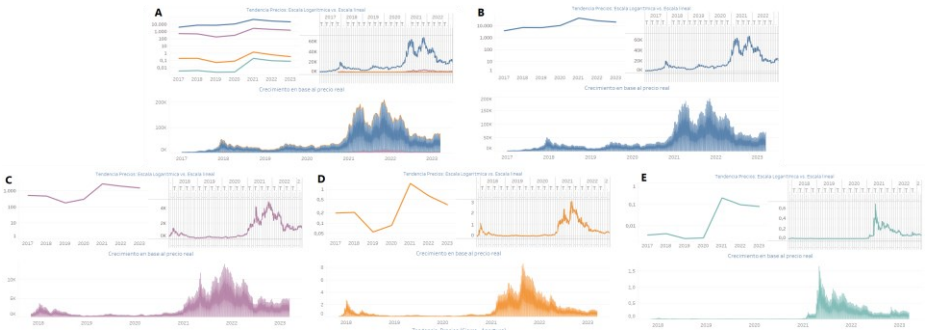


Fig.3. Trends of the four cryptocurrencies using line and area charts for the closing price. View (A) displays the trends of all cryptocurrencies together, while views (B), (C), (D), and (E) show the behavior of BTC, ETH, ADA, and DOGE respectively by utilizing filters in the *PDash*



Fig.4. Behavioural pattern detected in the *difference view* in the *PDash*. Periods of gains are highlighted in green, and losses are marked in red.



Fig.5. Line chart in the *VDash*. In **A** it can be observed that the growth rate of BTC's transaction volume is significantly higher than the rest of the cryptocurrencies, with all of them reaching their peak in volume in the year 2021. Applying quarterly temporal granularity, in **B**, the trend of BTC (blue) and ETH (purple) is analyzed.

For transaction volume trends detection **(Q3)**, the *VDash* was used. Figure 5A shows that from 2016, the transaction volume of BTC and ETH begins to grow significantly until the year 2021. In that year, all four cryptocurrencies reach their peak in transaction volume growth. However, after this peak, there is a period of considerable declines in the transaction volume for all four cryptocurrencies, while still maintaining an overall bullish trend. In the same chart, it is also evident that BTC has the highest transaction volume, followed by ETH, both of which are significantly higher than the other two cryptocurrencies. When changing the temporal granularity of the line chart, the transaction volume of BTC and ETH can be observed to be quite unstable, as shown in figure 5B. On the other hand, in figure 6A, it can be observed that the same pattern repeats for BTC during the second quarter of 2020 and the first and second quarters of 2021. This pattern shows a significant rise and then a steep decline in the average transaction volume. A very similar pattern can be seen in ETH during the second quarter of 2021, as shown in figure 6B. One of the reasons for these spikes could have been caused by the announcement from Elon Musk, the CEO of electric vehicle company Tesla, stating that his company would no longer accept BTC as payment due to the significant environmental cost of mining the cryptocurrency [15]. This not only affected BTC but also other cryptocurrencies. Another event that may have impacted the value of BTC during the same period was the restrictions imposed by China on cryptocurrencies [16].

To detect periods during which the currencies remain stable or exhibit an upward or downward trend **(Q4)** we use the *PDash*. Figure 7 shows in green the quarters (Q4 of 2017 and Q4 of 2020) in which bullish behaviors occurred, while in red, it displays the time periods (Q4 of 2018, Q4 of 2019, and Q2 of 2022) that presented more representative bearish behaviors considering the 4 cryptocurrencies. One of the factors that could have affected this behavior in May 2022 is the so-called *crypto winter* [17], which is associated with difficult moments in the global economy, where large capitals prefer to seek refuge in safer investments, causing a significant decline in the value of cryptocurrencies.

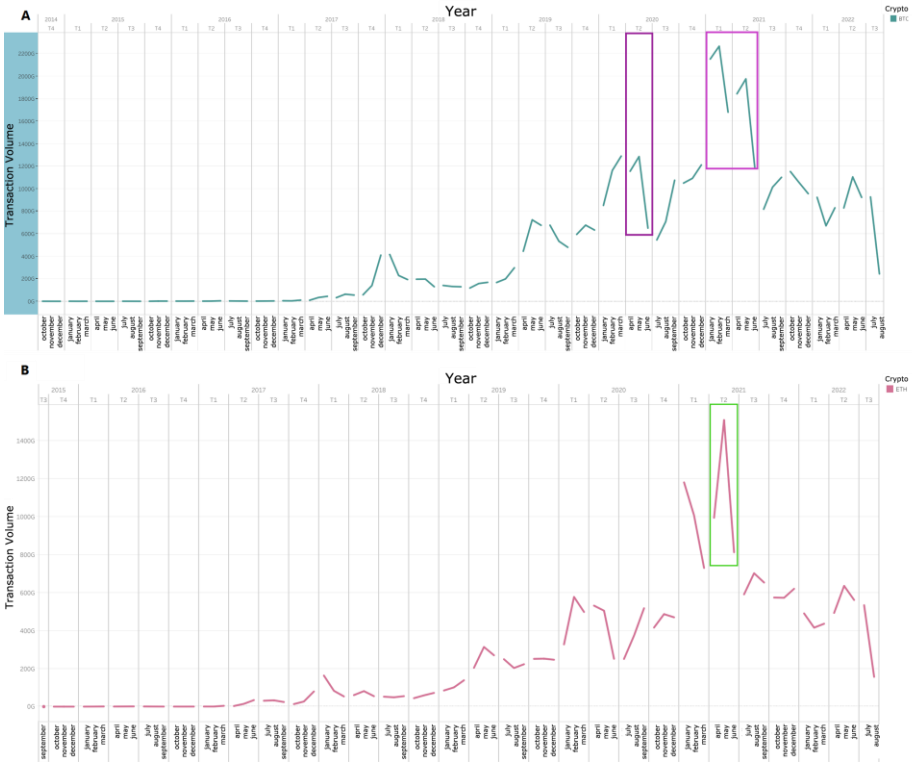


Fig.6. Line chart in the *VDash*. In **A**, purple rectangles highlight a behavioral pattern characterized by 3 periods of sharp declines for BTC volumes. **B** shows a strong increase in ETH's volume followed by a significant decline during the second quarter of 2021 with monthly granularity.

4. Discussion

The proposed solution and its exploration not only allowed us to answer our initial questions but also gave rise to new inquiries, such as: How volatile are the cryptocurrencies? **(Q5)**, Is there any daily seasonality regarding the transaction volume of cryptocurrencies? **(Q6)**, Is there any relationship between the closing price of each cryptocurrency and its transaction volume? **(Q7)**, among others.

When referring to volatility **(Q5)**, we are talking about the frequency and intensity of price changes of a stock or asset in a specific time range. This value is typically influenced by real events but also by speculation. Figure 8 displays ADA, DOGE, and ETH with very little variation; however, the blue highlighted area associated with BTC stands out with very abrupt changes, indicating high volatility in its price. This allows the user to understand the risks involved in investing in this currency, while also showing the behavior of other currencies that have remained more stable since their

inception. In this way, the intention is to provide the user with a general overview and enable them to make the best decision when investing in any cryptocurrency.

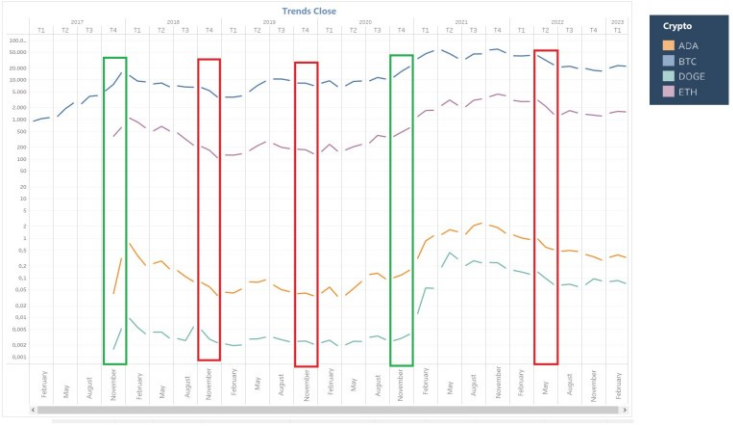


Fig.7. Line graph in the *PDash*. Periods of significant bullish behaviors (green rectangles) and bearish behaviors (red rectangles) considering the 4 cryptocurrencies.



Fig.8. *PDash*. High volatility of BTC compared to other cryptocurrencies.

Regarding **Q6**, a similar question is posed in [18], seeking to answer whether there is a "Monday effect" or "January effect" in financial markets. These are phenomena in which certain days of the week or months of the year have significantly higher or lower values compared to other periods. Figure 9A shows 4 bar charts, each corresponding to one of the cryptocurrencies. In each chart, it can be observed that Saturdays and Sundays are the days with the lowest number of transactions, while, for example, for

BTC (blue bars) and DOGE (bluish-green bars), Friday is the day with the highest number of transactions.



Fig.9. Volumen Dashboard. **B** Bar graph showing the relation between the day of the week and transaction volume. **B** Theme River showing the relation between price and transaction volume.

Regarding the relation between price and volume **(Q7)**, intuitively and based on the analysis of the presented dashboards, it is expected that both variables are related. If we observe figure 9B, we can conclude that, in general terms, the increase or decrease in the closing price of a cryptocurrency impacts its transaction volume. The graph shows that the peaks of both variables occur in very similar periods. Similarly, the periods of decline are well-marked.

5. Conclusions and Future Work

This paper presents a visualization tool for analyzing financial data of cryptocurrencies. The review of existing literature revealed that there are numerous visualization techniques associated with financial markets and cryptocurrencies in particular. However, to the best of our knowledge, there were no visualization tools that integrate multiple comparisons between more than one cryptocurrency in a single dashboard. In this context, we focused on the design and implementation of a visual analysis tool that allows the user to explore the trends and relations between different cryptocurrencies and answer important questions for the application domain. Future work includes collaboration with domain experts, implementing *CryptoVisualizer* in a lower-level language, and automating pattern detection with intelligent interactions.

Acknowledgments. This work was partially supported by PGI 24/N048 and PGI 24/ZN38 research grants from the Secretaría General de Ciencia y Tecnología, Universidad Nacional del Sur (Argentina), and by 28720210100824CO (PIBAA) granted by National Council for Scientific and Technical Research (CONICET).

References

1. Isabel Meirelles. Design for information: An introduction to the histories, theories, and best practices behind effective information visualizations, 2013.

2. Tamara Munzner and Eamonn Maguire. *Visualization analysis and design*. A K Peters visualization series. CRC Press, Boca Raton, FL, 2015.
3. Wolfgang Aigner, Silvia Miksch, Heidrun Schumann, and Christian Tominski. *Visualization of Time-Oriented Data*. Springer Publishing Company, Incorporated, 1st edition, 2011.
4. Wolfgang Aigner, Silvia Miksch, Wolfgang Müller, Heidrun Schumann, and Christian Tominski. Visualizing time-oriented data—a systematic view. *Computers & Graphics*, 31(3):401–409, 2007.
5. Marc Weber, Marc Alexa, and Wolfgang Müller. Visualizing time-series on spirals. In Keith Andrews, Steven F. Roth, and Pak Chung Wong, editors, *IEEE Symposium on Information Visualization 2001 (INFOVIS'01), San Diego, CA, USA, October 22-23, 2001*, pages 7–14. IEEE Computer Society, 2001.
6. Christian Tominski, James Abello, and Heidrun Schumann. Axes-based visualizations with radial layouts. In *ACM Symposium on Applied Computing*, 2004.
7. Natalia V. Andrienko and Gennady L. Andrienko. Exploratory analysis of spatial and temporal data - a systematic approach. Springer, 2006.
8. Julie Rodriguez and Piotr Kaczmarek. *Visualizing financial data*. John Wiley & Sons, 2016.
9. Coinmarketcap. https://coinmarketcap.com/es/. online July 2023.
10. Coingecko. https://www.coingecko.com/es. online July 2023.
11. Guang Sun * Jing Wang Ziyi Guo Yi Zuo Xiaoping Fan Jianjun Zhang Lin LangYue Cai, Zeying Song. On visualization analysis of stock data. *Journal on Big*
12. *Data*, 1(3):135–144, 2019.
13. Prasoon Kottarathil. https://es.finance.yahoo.com/criptomonedas/. online July
14. 2023.
15. Tableau software. https://www.tableau.com/. Consultado en Julio de 2023.
16. S. Havre, B. Hetzler, and L. Nowell. Themeriver: visualizing theme changes overtime. In *IEEE Symposium on Information Visualization 2000. INFOVIS 2000. Proceedings*, pages 115–123, 2000.
17. Español News. Dogecoin se recupera, bitcoin se hunde a medida que las criptomonedas suben y bajan en el viaje salvaje del Sr. Musk.
18. https://espanol.news/dogecoin-se-recupera-bitcoin-se-hunde-a-medida-que-lascriptomonedas-suben-y-bajan-en-el-viaje-salvaje-del-sr-musk/, 14 de mayo de 2021. online July 2023.
19. BBC News Mundo. La "bofetada" de China a las criptomonedas que hizo desplomarse al bitcoin. https://www.bbc.com/mundo/noticias-57165797, 19 de mayo de 2021. online July 2023.
20. Ámbito. ¿Terminó el cripto invierno? Esto dicen los analistas. https://www.ambito.com/finanzas/criptomonedas/termino-el-cripto-inviernoesto-dicen-los-analistas-n5487015, 2022. online on July 2023.
21. Aurelio F. Bariviera and Jorge de Andrés Sánchez. ¿existe estacionalidad diaria en el mercado de bonos y obligaciones del estado? evidencia empírica en el periodo 1998-2003. *Análisis Financiero*, 98:16–21, 2005.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Detección de daños con visión artificial en inspecciones marítimas: un mapeo sistemático de la literatura

Hugo Daniel Flores, Carlos Neil

Universidad Abierta Interamericana, Facultad de Tecnología Informática,
Centro de Altos Estudios en Tecnología Informática, Buenos Aires, Argentina
hugodaniel.flores@alumnos.uai.edu.ar; Carlos.Neil@uai.edu.ar

Resumen. *Contexto:* un alto porcentaje de vehículos son transportados vía marítima, donde son manipulados según protocolos específicos. Controlar su estado y detectar posibles daños recae en personal capacitado, lo que genera dificultades al momento de imputar responsabilidades por daños. Para mejorar la eficiencia de estos procesos, la visión artificial se ha consolidado como una alternativa viable para detectar y clasificar daños en distintas industrias. *Objetivos:* analizar el uso de algoritmos de inteligencia artificial para detectar y clasificar daños en la industria marítima. *Método:* se describe detalladamente la creación y ejecución de un protocolo que establece un conjunto de preguntas, así como el procedimiento para realizar la búsqueda bibliográfica y aplicar filtros para identificar los artículos relevantes. Luego, se realiza un análisis profundo para responder las preguntas planteadas en el protocolo. *Resultado:* se ha demostrado que las técnicas de visión artificial basadas en Redes Neuronales Convolucionales son altamente efectivas para implementar soluciones destinadas a la detección y clasificación de daños en esta industria. *Conclusión:* no se han identificado desarrollos tecnológicos similares para detectar y clasificar daños en transporte marítimo automotriz, por lo tanto, es viable explorar alternativas que incluyan técnicas de visión artificial para resolver los problemas planteados en la industria naviera.

Keywords: Daños, Redes Neuronales, Visión Artificial

1. Antecedentes

La producción de daños en unidades automotrices es un tema recurrente en la industria marítima y en particular en la industria automotriz. Los sistemas actuales procesan la información con los lineamientos convencionales relacionados con la aplicación de códigos alfanuméricos para especificar las zonas, las averías y la dimensión de ellas; y desde hace algunos años a esos códigos se han sumado las capturas de imágenes para certificar visualmente los daños. Como resultado de una inspección se generan reportes electrónicos para las empresas que intervienen en un punto de cambio de responsabilidad. Los problemas se presentan cada vez que se detecta un daño y este debe ser imputado a una empresa, lo cual muchas veces no es claro porque el daño puede haber sido provocado no por esa empresa sino por otra empresa en un punto de tránsito previo [1]. Identificar daños sirve para tomar medidas preventivas y también permite definir una estructura de costos para las reparaciones.

Asimismo, existen problemas relacionados con los daños ocurridos en lugares donde la vista humana no llega como pueden ser lugares bajos, o casos donde la luz o las condiciones del tiempo no son favorables para la visión humana. Los lugares donde

se presentan los problemas son en los cambios de responsabilidad: subida a buque, bajada de buque o dentro del buque. Además, es probable encontrar daños en ingreso o salida de puerto; y también pueden producirse daños en estiba en puerto. La importancia del tema en cuestión radica en que la evaluación actual de daños depende de la visión humana y del criterio y experiencia del perito, y los errores resultan sumamente costosos. También se presentan inconvenientes por las demoras en la emisión de informes en tiempo y forma. Toda información relacionada con un daño debe ser documentada para fabricantes, seguros, transportes, puertos y operadores logísticos. Demoras en los procesos administrativos produce demoras en la entrega de productos y pérdidas a las empresas. La utilidad de la solución propuesta produciría mejoras en todo el proceso de control de daños tanto directamente en campo de peritaje, como así también en la búsqueda de información en las bases de datos a través de las interfaces correspondientes. Por lo tanto, la inteligencia artificial (IA) en el procesamiento de imágenes daría mayor precisión en tiempo real y mejoraría la calidad de la información consultada. La utilización de tecnología de smartphones para la captura de imágenes y posterior análisis es de amplia utilización en la mayoría de los procedimientos de inspección. Actualmente toda información colectada es almacenada en bases datos. O sea que el procesamiento inteligente de imágenes sería una herramienta ampliamente utilizada en la automatización de estos procesos industriales, debido a que presenta confiabilidad, eficacia y rapidez en su proceso.

Por lo expuesto, se propone la automatización de inspecciones para detectar y clasificar daños usando visión artificial en dispositivos móviles. El sistema se basaría en un clasificador de imágenes desarrollado con técnicas de aprendizaje supervisado con redes neuronales convolucionales, una de las arquitecturas de aprendizaje profundo más populares, diseñadas para resolver problemas de visión artificial, reconocimiento de patrones y clasificación de imágenes. En la actualidad las inspecciones se desarrollan con celulares y aplicaciones a medida que permiten codificar los daños encontrados y tomar fotos. Hoy en día no se cuenta con información acerca de una herramienta inteligente que permita a los inspectores resolver la problemática de detección de daños. La utilización de celulares en los procesos de revisión con captura de imágenes brinda una alternativa viable para un desarrollo tecnológico móvil inteligente. Además, también se podría considerar la posibilidad de un desarrollo informático para el procesamiento digital de imágenes utilizando los canales convencionales de ingresos de datos.

Para concluir, este trabajo tiene como objetivo realizar un mapeo sistemático de la literatura, con el cual se podrá realizar un análisis de los artículos que incluyan temas acerca de los algoritmos que permiten el reconocimiento y clasificación de daños con visión artificial para diferentes tipos de problemáticas.

2. Preguntas de investigación

Como primera etapa de este mapeo sistemático se definieron las preguntas (P) de investigación siguientes y sus correspondientes motivaciones (M):

Tabla 1: Preguntas guías del mapeo sistemático de la literatura.

PREGUNTAS INVESTIGACION	MOTIVACIÓN
P1: ¿Cuáles son los principales problemas abordados en la detección de daños usando imágenes tomadas desde un dispositivo móvil?	M1: Especificar procesos de inspección donde se brinda solución a problemas de imputabilidad de daños en tiempo real. Reducir costos en proceso de control. Identificar daños en partes de difícil acceso sea por la zona en particular del auto o por las condiciones climáticas.
P2: ¿Qué tipos de imágenes se usan en estudios relacionados con la detección de daños usando imágenes desde celulares?	M2: Determinar qué tipo de herramientas tecnológicas se utilizan en la captura de imágenes y los formatos y tamaños de archivos con los que se realiza el proceso de identificación de daños.
P3: ¿Cuáles son las metodologías usadas en la inteligencia artificial para ayudar a identificar daños en el transporte de autos vía marítima?	M3: Encontrar propuestas referidas a la detección de daños en autos 0 KM que permitan corroborar que la utilización de inteligencia artificial en esta industria es viable.
P4: ¿Cuál es el nivel de madurez de las metodologías encontradas?	M4: Mostrar cuales son los avances de la visión por computadoras en los últimos 12 años.
P5: ¿Cuántas publicaciones se han llevado a cabo durante los últimos 12 años con respecto a la detección de daños en la industria automotriz?	M5: Especificar las cantidades estimadas de investigaciones realizadas en visión artificial en procesos industriales de inspección marítima de autos en los últimos 12 años.
P6: ¿Cuáles son los algoritmos más usados en cada categoría?	M6: Identificar los algoritmos y los enfoques más usados en visión artificial para detectar y clasificar daños en el transporte marítimo de autos 0 KM.

3. Método de revisión

Este trabajo fue realizado con una metodología de Mapeo Sistemático de la Literatura que permite hacer una búsqueda amplia de información sobre uno o varios temas específicos a través de artículos científicos. Como primer paso se identificó una problemática la cual se convirtió en un tema de investigación y discusión bien sustentada del estado del arte. También se combinaron diferentes facetas del esquema para responder preguntas de investigación más específicas. Una vez terminada la búsqueda se realizó la conclusión de los resultados logrados. Siguiendo el proceso de mapeo sistemático propuesto por [2] se trabajó en esta búsqueda a partir de las 4 etapas básicas: Especificar Preguntas de Investigación, Definir Protocolos de Revisión, Ejecutar Búsquedas y finalmente la Conclusión del Estudio.

3.1 Fuentes

Para obtener la totalidad de los artículos para este mapeo sistemático de la literatura se utilizaron fuentes electrónicas masivas, es decir bases de datos que exponen documentos de diversos temas de investigación. Esto se decidió así debido a que no

existen repositorios o webs especializadas en la temática aquí tratada. Por este motivo se seleccionaron algunas de las más populares y pobladas fuentes de artículos relacionados con la aplicación de algoritmos utilizados para detectar y clasificar daños con visión artificial, como lo son: SEDICI, IEEE Xplore Digital Library, ACM Digital Library y Science Direct.

3.2 Definición de términos

Siguiendo la metodología propuesta, se definió una cadena de búsqueda (CB) que permitió obtener artículos que se utilizaron para realizar un mapeo sistemático. Primero, se creó la cadena de búsqueda basándose en tres términos principales (TP): CB1 (computer vision AND neural network AND damage detection). Si bien la búsqueda realizada con la aplicación de los términos principales arrojó los resultados esperados se pudo ampliar la lista de artículos con la implementación de los términos alternativos (TA): CB2 (computer visión OR artificial intelligence) AND (neural network OR convolutional neural network OR deep learnig OR machine learning) AND (damage detection). Como consecuencia que el desarrollo informático propuesto es para la utilización de una tecnología aplicada a la solución de problemas en una industria específica, se ha incluido entre las palabras principales “damage detection” y eso ha reducido en gran medida la lista de artículos encontrados.

3.3 Criterios de inclusión y exclusión

Luego de ejecutados los comandos de búsquedas se procedió a aplicar los criterios de inclusión (CI) y exclusión (CE). El paso necesario una vez obtenidos los artículos de las bases de datos es seleccionar los que aportan información para responder a las preguntas planteadas en el trabajo y descartar el resto. Este proceso fue iterativo y por cada artículo se aplicaron las diferentes reglas de selección. Cabe destacar que en este punto se seleccionaron un conjunto de artículos relevantes pero que fueron analizados en profundidad en la próxima sección cuando se aplican los filtros que son basados en los criterios aquí establecidos. Para completar la selección se aplican CI y CE:

Tabla 2: Criterios de inclusión y exclusión.

CRITERIOS DE INCLUSION
CI1: Estudios en inglés y español. publicados en el periodo comprendido entre 2010 y 2022.
CI2: Estudios relacionados con la detección y clasificación de daños.
CI3: Campo de investigación: ingeniería informática.
CI4: Tipo de documento: documento de conferencia y artículo.
CI5: Artículos científicos que se relacionen directamente con la detección y clasificación de daños y utilicen algoritmos de IA, y que tengan como enfoque la visión por computadora.
CRITERIOS DE EXCLUSION
CE1: Estudios que no se basen en el uso de imágenes, publicados previamente al 2010.
CE2: Se descartan las publicaciones que no usen algoritmos para la identificación, clasificación y reconocimiento de patrones de imágenes.

4. Búsqueda de trabajos

Para la obtención de los artículos se siguió un procedimiento de cuatro pasos. El punto de inicio fue la definición de las preguntas de investigación, creación de la cadena de búsqueda y de los CI y CE para la selección de los artículos. Posteriormente se realizó la búsqueda de trabajos en cada una de las bases de datos dando por resultado SEDICI: 197 art., IEEE Xplore Digital Library: 106 art., ACM Digital Library: 20 art., y Science Direct: 186 art. Luego, sobre esta cantidad de trabajos se aplicaron CI y CE. Este paso fue iterativo y constó de dos etapas: primer filtro y segundo filtro. Para el primer filtro se analizaron el título, abstract y keywords quedando 91 art. En segunda instancia también se tuvo en cuenta la introducción y la conclusión dando por resultado un total de 41 art. que son los utilizados para este trabajo. Además, dentro de la clasificación de artículos interesantes para este mapeo sistemático se fueron identificando los estudios relacionados con la problemática central de daños en autos y por otro lado los trabajos referidos específicamente a la utilización de Redes Neuronales Convolucionales (RNC). Finalmente, para profundizar en el resultado de esta revisión se han incluido artículos donde se aplican más de una técnica de inteligencia artificial para detectar y clasificar daños.

5. Síntesis de datos extraídos

La búsqueda y selección de trabajos permitió obtener un conjunto que representa la tendencia dentro del tema investigado en el periodo establecido en el protocolo. Esto facilitó la extracción de información, identificando técnicas y tecnologías más usadas con el fin de responder a las preguntas de investigación. A continuación, se exponen sintéticamente los resultados de este mapeo sistemático de la literatura.

P1: ¿Cuáles son los principales problemas abordados en la detección de daños usando imágenes tomadas desde un dispositivo móvil? Optimizar procesos de inspección y detectar y clasificar daños en tiempo real son los objetivos para cumplir por las diferentes industrias en los textos analizados, y al igual que la reducción de los costos; estos son logrados con la utilización de visión por computadora. En función de los resultados principales obtenidos por este estudio se puede inferir que los problemas que abordan la detección de daños con imágenes tomadas por celulares se encuentran en las industrias de transporte [3], [4], [5] y [6] y de alimentación [7]. Mientras que otros tipos de estudios que han utilizado redes neuronales para la detección de daños son abordados en los servicios de seguros [8], [9], [10], [11], [12], [13], [14] y [15], en alimentación [16], en medio ambiente [17], en comunicación [18], en construcción [19] y en la industria automotriz [20]; todos ellos han utilizado buscadores de internet y repositorios para entrenar los modelos propuestos. Y para seguir con el análisis de los resultados sobre los problemas abordados utilizando dispositivos digitales específicos (cámaras, filmadoras, etc.) para la captura de imágenes se pueden detallar en la industria del transporte [21], servicios de seguros [22], en alimentación [23], [24], [25], [26], [27], [28], [29], [30], [31] y [32], en medio ambiente [33], [34] y [35], en la industria de la construcción [36], en medicina [37] y [38], y en la industria automotriz [39]. Finalmente, para concluir con este ítem se pueden mencionar los artículos donde no se han realizado especificaciones acerca de los recursos utilizados para el entrenamiento

de los algoritmos: industrias del transporte [40], alimentación [41], paleontología [42] y en la industria automotriz [43].

P2: ¿Qué tipos de imágenes se usan en estudios relacionados con la detección de daños usando imágenes desde celulares? Más del 50% de los estudios primarios contaron con sistemas de captura digital como cámaras, filmadoras o sistemas específicos de toma de imágenes o buscadores y repositorios en línea. Cuatro estudios utilizaron smartphones para la captura de imágenes. Y solo hay 1 estudio donde se pudo ver que se usaron tanto celulares como archivos descargados de internet para lograr los objetivos propuestos. Del análisis de las fuentes de producción de imágenes, se puede deducir que de los 41 estudios primarios obtenidos 13 utilizaron bases de datos de buscadores y repositorios online y 19 artículos mostraron como fuente de datos sistemas digitales como cámaras, filmadoras y equipos especiales. Y finalmente se pudo verificar que 4 trabajos no hacen referencia a las fuentes con las que trabajaron en sus respectivos proyectos.

P3: ¿Cuáles son las metodologías usadas en la inteligencia artificial para ayudar a identificar daños en el transporte de autos vía marítima? Puntualmente este tipo de metodología para detectar daños en la industria marítima en el transporte de autos no se aplica en la actualidad. Aunque si se han encontrado dentro de este estudio técnicas de inteligencia artificial para resolver el problema de la detección de daños con visión artificial en autos. Estos artículos se refieren a la producción e identificación de daños en servicios de seguros y de transportes. A continuación, se detallan los artículos:

Tabla 3: Técnicas de inteligencia artificial que aportan soluciones a esta temática.

ART.	SOL.	APORTE
[8]	RNC	Evaluación de daños en autos con redes neuronales convolucionales para detectar daños destinado especialmente a empresas aseguradoras.
[9]	RNC	Modelo de inspección de vehículos automático usando un enfoque de aprendizaje profundo para la agilización de procesos de reclamos de seguros.
[10]	RNC	Clasificador de daños en autos utilizando un modelo de aprendizaje profundo para optimizar procesos de reclamos de seguros.
[11]	RNC	Detección y análisis de daños en autos usando algoritmos de aprendizaje profundo destinado a los reclamos de seguros en forma oportuna.
[12]	RNC	Detección de daños en autos usando aprendizaje profundo para tramitación de seguros por accidentes de tránsito.
[13]	RNC	Evaluación de daños en autos para mejorar procesos de reclamos en seguros.
[14]	RNC	Prevención de daños usando redes neuronales convolucionales y visión por computadora para facilitar reclamos de seguros por accidentes de tránsito.
[22]	RNC	Reconocimiento de superficies de partes de autos dañados en tiempo real usando una herramienta de aprendizaje profundo destinado a la evaluación de datos para reclamo de seguros.
[15]	YOLO	Detección y clasificación de daños en vehículos usando procesamiento de imágenes para automatización de reclamos de proveedores de seguros.
[7]	RNC	Redes neuronales convolucionales para la detección de daños en vehículos destinado al sistema de transporte con movilidad compartida.

P4: ¿Cuál es el nivel de madurez de las metodologías encontradas? Se presentan grandes avances en la tecnología de visión por computadora. Los campos en

los que más se encontraron trabajos de investigación relacionados con esta problemática es la de servicios de seguros (9 art.) donde son ampliamente utilizadas las RNC y YOLO. En el área de la alimentación (13 art.) también hay utilización de RNC y YOLO, pero además se suman la utilización de técnicas de Redes Neuronales Artificiales (RNA) solamente, y también se muestran trabajos de aplicaciones comparativas entre RNC y Vectores de Fisher o Máquinas de Soporte Vectorial (MSV). Los estudios se refieren a identificación de productos, detección de enfermedades de cultivos y 1 particular aplicado al análisis de cortes de carne vacuna. Otra área donde también se utilizan RNC y YOLO es en el de transporte (6 art.) y también se encuentran algunas aplicaciones de RNA solamente. Aquí los estudios también tienen por objetivo solucionar problemas de detección de daños, y además se utilizan estas tecnologías para el reconocimiento de patentes de autos. En cuanto a las industrias de la construcción (2 art.) y las comunicaciones interpersonales (1 art.) se puede ver la utilización de RNC para dar respuesta a las problemáticas asociadas a la detección de roturas de hormigón y el intercambio de mensajes de señas. En medio ambiente (4 art.) se puede observar la utilización de RNC, RNA, MSV, Red Neuronal Feed-Fordward Backpropagation y Red Bayesiana para el cuidado de bosques, calidad de fibras textiles y prevención de incendios forestales. Y en medicina y paleontología (3 art.) se ve el uso de RNC, MSV, RNA y Redes Bayesianas. En medicina el objetivo era detectar enfermedades y en paleontología tenían como propósito dar soporte para el posicionamiento de partes fósiles. Finalmente, en la industria automotriz se observan 3 art. donde se usan RNC para detectar y clasificar daños, MLP (MultiLayer Perceptron) para hacer medición entre vehículos en circulación, y un sistema de barrido de iluminación construido a la salida de fábrica de producción de autos para detectar daños en diferentes zonas.

P5: ¿Cuántas publicaciones se han llevado a cabo durante los últimos 12 años con respecto a la detección de daños en la industria automotriz? A partir del año 2015 se muestra un incremento notorio de las investigaciones realizadas en visión por computadora en procesos industriales. En el caso de publicaciones para solucionar problemas de la industria automotriz con la detección de daños se encontraron 1 estudio en 2012 [43] y 2 estudios entre 2019 [39] y 2020 [20]; pero esa cantidad de trabajos podría ser ampliada si se consideran los trabajos de servicios de seguros para resolver problemas relacionados con autos y la producción de daños, o sea que podrían sumarse 9 estudios más relacionados con esa industria (4 en 2019 [10], [11], [12], [14]; 1 en 2021 [8] y 4 en 2022 [9], [13], [7], [22]).

P6: ¿Cuáles son los algoritmos más usados en cada categoría? En el enfoque de visión por computadora entre los algoritmos más usados para esta investigación se encontraron los algoritmos CNN (Convolutional Neural Network, 56,10%), ANN (Artificial Neural Network, 17,07%), YOLO (You Only Look Once, 7,32%) y CNN junto a Support Vector Machines (4,88%). Finalmente, en la misma proporción CNN más Vectores de Fisher (2,44%), ANN más SVM (2,44%), Red Neuronal Feed-Fordward Backpropagation, Red Bayesiana y SVM (2,44%), ANN más Redes Bayesianas (2,44%), MLP (2,44%) y Barrido de iluminación (2,44%). En el rubro que interesa a este trabajo de investigación el algoritmo más usado en la industria automotriz para clasificar y detectar daños en diferentes zonas es el de CNN, 56,10%. Si bien los artículos encontrados no se refieren directamente a la solución de la problemática planteada en el transporte de autos por buque, si es importante mencionar que esta revisión de la literatura ha dado una cuota de seguridad para iniciar un proyecto con la

aplicación de RNC para resolver los problemas referidos a los procesos de inspección inteligentes en la industria marítima.

6. Conclusiones

Se han logrado presentar los resultados de un estudio de mapeo sistemático para revisar el estado del arte de la inteligencia artificial en la industria marítima. Si bien el estudio ha cumplido las expectativas no ha logrado evidenciar desarrollos tecnológicos de estas características en la industria automotriz específicamente en el transporte marítimo, pero ha servido para concluir que el aprendizaje supervisado con técnicas de visión artificial podría dar respuesta a la detección de daños con dispositivos móviles en la industria marítima y en este caso puntual al transporte de autos. Si bien sirvió este estudio para justificar la utilización de RNC para clasificar y detectar daños y su responsabilidad, también ha sido de mucha utilidad relevar datos que muestran el bajo número de desarrollos tecnológicos en la industria de autos, específicamente en la logística necesaria para distribuir la producción. Entre los principales problemas abordados en la detección de daños usando imágenes tomadas desde un dispositivo móvil se pueden mencionar los referidos a la optimización de procesos de inspección, la imputación en tiempo real de daños, y la reducción de costos por reparaciones como así también la identificación de daños en sectores de difícil acceso y bajo condiciones de trabajo adversas.

Por último, la captura de imágenes para un desarrollo tecnológico en función de esta revisión de la literatura puede ser viables a través de smartphones debido a la madurez que ha tenido esta tecnología en los últimos años y a la versatilidad de los algoritmos de inteligencia artificial para procesar las características de las imágenes tomadas. Se presentan grandes avances en la tecnología de visión por computadora en los últimos 12 años y a partir del año 2015 se muestra un incremento notorio de las investigaciones realizadas. En el enfoque de visión por computadora entre los algoritmos más usados se encontraron los algoritmos CNN, ANN, YOLO y SVM.

Referencias

1. H. Flores, R. Garcia Martinez, E. Fernandez, H. Merlino, D. Rodrguez, and P. Britos: Detección de Patrones para la Prevención de Daños y/o Averías en la Industria Automotriz. In XV Congreso Argentino de Ciencias de la Computación, 2009, pp. 1021-1030.
2. Barbara Kitchenham, Rialette Pretorius, David Budgen, O. Pearl Brereton, Mark Turner, Mahmood Niazi and Stephen Linkman: Systematic literature reviews in software engineering - A tertiary study, In Information and Software Technology, Vol. 52 pp. 792–805. Elsevier, March 2010.
3. L. Zhang, F. Yang, Y. Daniel Zhang and Y. J. Zhu: Road crack detection using deep convolutional neural network. In Proceedings International Conference on Image Processing ICIP, vol. 2016 August, pp. 3708–3712, August 2016.
4. K. Doshi and Y. Yilmaz: Road Damage Detection using Deep Ensemble Learning. In Proceedings 2020 IEEE International Conference on Big Data, Big Data 2020, pp. 5540–5544, December 2020.

5. R. Vishwakarma and R. Vennelakanti: CNN Model Tuning for Global Road Damage Detection, Proceedings 2020 IEEE International Conference on Big Data, Big Data 2020, pp. 5609–5615, December 2020.
6. L. B. Córdova and M. P. Periche: Detección de plaga cogollero (Spodoptera frugiperda) mediante un sistema de reconocimiento de imagen en cultivo de maíz, Tambogrande Piura 2021. B.S. thesis, Universidad Cesar Vallejos, Piura, Perú, 2022. June 05, 2022.
7. R. E. van Ruitenbeek and S. Bhulai: Convolutional Neural Networks for vehicle damage detection. In Machine Learning with Applications, vol. 9, p. 100332, September 2022.
8. H. Bandi, S. Joshi, S. Bhagat, and A. Deshpande: Assessing Car Damage with Convolutional Neural Networks. In Proceedings International Conference on Communication, Information and Computing Technology, ICCICT 2021, 2021.
9. M. M. Fouad, K. Malawany, A. G. Osman, H. M. Amer, A. M. Abdulkhalek and A. B. Eldin: Automated vehicle inspection model using a deep learning approach. In J Ambient Intell Humaniz Comput, 2022.
10. D. Parra and A. E. Gonzalez: Car Damage Assesment. In College of Engineering and Computer Science UTRGV Edinburg, Texas, USA, 2019.
11. Rakshata P, Padma H V and Pooja M: Car Damage Detection and Analysis Using Deep Learning Algorithm For Automotive. In International Journal of Scientific Research & Engineering Trends, Volume 5, Issue 6, Nov-Dec-2019, ISSN: 2395-566X, 2019. December 20, 2022.
12. M. Aqeel Arshad: Car Damage Detection using Machine Learning. In IJARCCE International Journal of Advanced Research in Computer and Communication Engineering, vol. 10, 2021.
13. V. Agarwal, U. Khandelwal, S. Kumar and R. Kumar: Damage Assessment of a vehicle and Insurance Reclaim. In 2022I JCRT, Volume10, Issue, April 2022, ISSN:2320-2882, 2022. December 20, 2022.
14. Sharma A., Verma A. and Gupta D.: Preventing Car Damage using CNN and Computer Vision. In International Journal of Innovative Technology and Exploring Engineering (IJITEE), ISSN: 2278-3075. Volume-9 Issue-1, November 2019.
15. A. K. Kumar L.: Vehicle Damage Detection and Classification Using Image Processing. In International Journal of Advanced Research in Science, Communication and Technology IJARSCT, vol. 2, no. 9, 2022.
16. N. Fabián, P. Fuentes, G.: Profesor, D. de Postgrado and U. de Concepción: Visión por computadora para el manejo de plagas y enfermedades en cultivos de papa. M.S. tesis, Universidad de Concepción, Concepción, Chile, 2021. June 05, 2022.
17. H. Kibriya, I. Abdullah and A. Nasrullah: Plant Disease Identification and Classification Using Convolutional Neural Network and SVM. In Proceedings 2021 International Conference on Frontiers of Information Technology, FIT 2021, pp. 264–268, 2021.
18. Marcos Joel Vásquez Soto: Aplicación móvil basado en el reconocimiento de imágenes para apoyar el aprendizaje del lenguaje de señas de gestos estáticos en el Perú. B.S. thesis, Universidad Católica Santo Toribio de Mogrovejo, Chiclayo, Perú, 2021. June 05, 2022.
19. P. Kumar, A. Sharma and S. R. Kota: Automatic Multiclass Instance Segmentation of Concrete Damage Using Deep Learning Model. In IEEE, vol. 9, pp. 90330–90345, 2021.
20. Neriman YARAŞ: Vehicle type classification whith deep learning. B.S. thesis, İzmir Institute of Technology, Izmir, Turquía, 2020.
21. Jun Lu, Bo Liang, Qujiang Lei, Xiuhao Li, Junhao Liu, Ji Liu, Jie Xu and Weijun Wang: SCueU-Net: Efficient Damage Detection Method for Railway Rail. In IEEE, vol. 8, pp. 125109–125120, 2020.
22. M. Parhizkar and M. Amirfakhrian: Recognizing the Damaged Surface Parts of Cars in the Real Scene Using a Deep Learning Framework. In Math Probl Eng, vol. 2022, 2022.

23. C. Lema Mercedes Azucena and Tumbaco Sango Jenny Celia: Aplicación móvil para clasificación automática de malas hierbas en imágenes usando aprendizaje profundo. B.S. thesis, Universidad Técnica de Cotopaxi, Latacunga, Ecuador, 2020. June 04, 2022.
24. S. Bobde, S. Jaiswal, P. Kulkarni, O. Patil, P. Khode and R. Jha: Fruit Quality Recognition using Deep Learning Algorithm. In 2021 International Conference on Smart Generation Computing, Communication and Networking, SMART GENCON 2021, 2021.
25. E. Ennadifi, S. Laraba, D. Vincke, B. Mercatoris and B. Gosselin: Wheat Diseases Classification and Localization Using Convolutional Neural Networks and GradCAM Visualization. In 2020 International Conference on Intelligent Systems and Computer Vision, ISCV 2020, June 2020.
26. Javier Berger, Alejandro Tamada and Rafael Feltan: Aplicación móvil para la identificación de variedades de Manihot esculenta Crantz cultivadas en Misiones mediante técnicas de deep learning. In 48JAIIO, Congreso Argentino de AgroInformática, 2022, ISSN: 25250949, pp. 104 – 115. July 29, 2022.
27. Ulises Bussi, Giselle Mazzarello, Marta Ochoa, Fabián Sanfilippo and Damián Oliva: Implementación de un sistema de análisis de cortes vacunos basado en redes neuronales. In ASAI 2015, 16º Simposio Argentino de Inteligencia Artificial, 2015. July 29, 2022.
28. Schneider Gerardo E., Solano Agustín, Kemerer Alejandra and Hadad Alejandro: Gestión y procesamiento de imágenes aéreas de caña de azúcar. In XVII Workshop de Investigadores en Ciencias de la Computación (Salta, 2015). July 29, 2022.
29. Favret, Fabián, Eckert, Karina, Felten, Analía and Sanderg, Gabriela: Reconocimiento de patrones mediante redes neuronales para la determinación del nivel de contenido de palo en la yerba mate. In XVII Workshop de Investigadores en Ciencias de la Computación (Salta, 2015). July 29, 2022.
30. N. L. Grisales: Identificación y conteo de insectos (mosca blanca, trips y minador de hoja) capturados en trampa adhesiva mediante visión e inteligencia artificial. Caso Flores El Trigal. B.S. thesis, Universidad EIA Ingeniería Mecatrónica, Envigado, Colombia, 2020. June 04, 2022.
31. Redolfi, Javier A., González Dondo, Diego, Pucheta, Julián Antonio and Canali, Luis R.: Clasificación de variedades de semillas de trigo usando visión por computadora. In VIII Congreso Argentino de AgroInformática (CAI-2016) - JAIIO 45 (Tres de Febrero, 2016), 2016. July 29, 2022.
32. N. E. Villafuerte: Diagnóstico automatizado para la clasificación de café trillado con broca mediante procesamiento de imágenes con Deep Learning. B.S. thesis, Universidad Peruana Unión, Perú, 2019. June 05, 2022.
33. Constable, Leticia, Arcidiácono, Marcelo, Vázquez, Juan C. and Picco, Juan: Tratamiento de Imágenes de Fibras Animales. In XV Workshop de Investigadores en Ciencias de la Computación, 2013. July 29, 2022.
34. Cardenas, Marina E., Medel, Ricardo, Castillo, Julio J., Vázquez, Juan C. and Casco, Osvaldo: Modelos de aprendizaje supervisados: aplicaciones para la predicción de incendios forestales en la provincia de Córdoba. In XVII Workshop de Investigadores en Ciencias de la Computación (Salta, 2015), 2015. July 29, 2022.
35. Davila Garcia and Jorge Rodrigo: Sistema experto con visión artificial para la identificación de insectos descortezadores en especies de pino de los estados forestales más importantes de México. M.S. thesis, 2019, Universidad Autónoma del Estado de México. June 05, 2022.
36. Colpari Pozzo and Mauricio Henry: Uso de inteligencia artificial para la detección automatizada de fisuras en estructuras de hormigón armado. M.S. thesis, Pontificia Universidad Católica de Chile, Escuela de Construcción Civil, Chile, 2020. June 05, 2022.
37. J. Islam and Y. Zhang: Early diagnosis of alzheimer's disease: A neuroimaging study with deep learning architectures. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, vol. 2018-June, pp. 1962–1964, December 2018.

38. N. Chakrabarty and S. Chatterjee: An Offbeat Technique for Diabetic Retinopathy Detection using Computer Vision. In 2019 10th International Conference on Computing, Communication and Networking Technologies, ICCCNT 2019, July 2019.
39. M. Amine and K. Djoudi: Vehicles detection using the MLP and the correlation measurement. In 2019 International Conference on Advanced Electrical Engineering, ICAEE 2019, November 2019.
40. Gazcón, Nicolás Fernando, Chesñevar, Carlos Iván and Castro, Silvia Mabel: Intelligent Template Matching: un nuevo acercamiento para reconocimiento automático de patentes argentinas. In XVII Congreso Argentino de Ciencias de la Computación, 2011. July 29, 2022.
41. Andres Camilo Rincon Caballero and Maryury Julietth Rodriguez Calderon: Iacultivar: aplicación móvil con reconocimiento de imágenes para la detección y prevención de la monilla en el cultivo del cacao. B.S. thesis, Universidad Autónoma de Bucaramanga, Bucaramanga, Colombia, 2020. June. 04, 2022.
42. Cintas Celia, Delrieux Claudio, Bianchi Gloria Susana and Defossé, Nahuel: Procesamiento inteligente de imágenes: Posicionamiento de vértebras mediante Landmarks y Redes Neuronales. In XIV Workshop de Investigadores en Ciencias de la Computación, 2012. July 29, 2022.
43. J. Tornero, L. Armesto, M. C. Mora, N. Montés, A. Herráez and J. Asensio: Detección de defectos en carrocerías de vehículos basado en visión artificial: Diseño e implantación. In Revista Iberoamericana de Automática e Informática Industrial (RIAI), vol. 9, pp. 93–104, 2012.

Coordinadores:
- Cecilia Sanz (UNLP)
- Beatríz Depetris (UNTDF)
- Marcelo De Vincenzi (UAI)

CACiC 2023

CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

XII Workshop de Innovación en Educación en Informática

> Full Papers

WIEI

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Estilos de aprendizajes personalizados con inteligencia artificial. Un mapeo sistemático de la literatura

Fabián Maffei, Carlos Neil, Nicolás Battaglia.

Universidad Abierta Interamericana. Facultad de Tecnología Informática.
Centro de Altos Estudios en Tecnología Informática. Buenos Aires. Argentina
Fabian.maffei@alumnos.uai.edu.ar
{Carlos.Neil, Nicolas.Battaglia}@uai.edu.ar

Abstract. **Contexto**: la personalización de la enseñanza y el aprendizaje sustantivo mediante inteligencia artificial (IA) para identificar estilos de aprendizaje. **Objetivo**: interpretar las investigaciones en IA con el propósito de mejorar el rendimiento académico y detectar posibles brechas en el conocimiento. **Método**: se llevaron a cabo preguntas de investigación, búsquedas y análisis de datos para estudiar los métodos de enseñanza que utilizan IA, los estilos de aprendizaje y los sistemas adaptativos. **Resultados**: se concluye que para maximizar el rendimiento académico, es crucial reconocer los patrones de estilos de aprendizaje y utilizar herramientas de IA para desarrollar agentes inteligentes personalizados en entornos virtuales de aprendizaje.

Keywords: Estilos de aprendizaje, inteligencia artificial, agentes inteligentes, sistemas adaptativos

1. Introducción

La formación de los estudiantes mejoraría notablemente si se incorporara tecnología que le permita al formador adaptar las estrategias de enseñanza a las características del aprendizaje del estudiante. Si bien es cada vez más frecuente el uso de plataformas LMS que permiten adoptar criterios y estrategias, estas acciones son propuestas por el docente en función de su formación, experiencia y criterios subjetivos. Así, no hay certezas en la identificación del estilo de aprendizaje eficiente para cada estudiante, basada en inteligencia artificial (IA), y así personalizar las estrategias de enseñanza [3]. Con el objetivo de realizar un mapeo sistemático de la literatura que permita establecer qué artículos de investigación desarrollaron esta temática, en este trabajo se pretende indagar en las siguientes cuestiones:

- ¿Qué etapas son necesarias para desarrollar un "Sistema de Evaluación Automática" (SEA) que recoja fortalezas y debilidades de los estilos de aprendizaje de estudiantes?
- ¿Cómo determinar las características necesarias que deberá poseer un sistema adaptativo (SA), con capacidades para ajustar su funcionamiento a la característica de un usuario o grupo de usuarios?
- ¿Cómo diseñar una aplicación educativa basada en inteligencia artificial (IA) que permita personalizar las estrategias de enseñanza, a través de un "sistema de evaluación automática" y su posterior "sistema adaptativo"?

Este trabajo se realiza en el marco del Proyecto de investigación UAI CASE "Herramientas Colaborativas Multiplataforma en la Enseñanza de la Ingeniería de Software", en desarrollo en la Facultad de ingeniería de la Universidad Abierta Interamericana, se propone la integración de plataformas tecnológicas para el aprendizaje ubicuo colaborativo, en cursos relacionados a la ingeniería de software, en particular, durante el proceso de modelado de software.

2. Preguntas de investigación

En esta primera etapa se define el problema a través de preguntas planteadas de investigación, que orientarán el mapeo sistemático de la literatura presente.

Tabla 1. Preguntas de investigación

Preguntas de investigación	Motivación
P1 ¿Qué estudios primarios existen sobre métodos de enseñanza, modelado con inteligencia artificial considerando los estilos de aprendizaje del estudiante?	M1. Revisar investigaciones sobre el uso de inteligencia artificial en la educación, enfocándose en métodos de enseñanza que consideren los estilos de aprendizaje de los estudiantes.
P2 ¿Cuáles métodos y técnicas son las más utilizadas para reconocer los diferentes estilos de aprendizaje de un estudiante?	M2. Explorar los métodos efectivos para identificar estilos de aprendizaje de los estudiantes, con el objetivo de personalizar la enseñanza y mejorar el rendimiento académico. Además, se busca identificar áreas de investigación con brechas y necesidades para mejorar la detección de estilos de aprendizaje
P3 ¿Qué atributos tienen los agentes inteligentes que han sido propuestos como soluciones?	M3. Analizar las características y propiedades de agentes inteligentes utilizados en sistemas educativos basados en inteligencia artificial, con el objetivo de evaluar su efectividad y adecuación en el contexto educativo, así como mejorar la calidad de la experiencia de aprendizaje del estudiante
P4 ¿Cuáles son los criterios de validación de un algoritmo de aprendizaje automático de un sistema de evaluación automático, que permitan la toma de decisiones con mínima intervención humana?	M4. Examinar los estándares y requisitos para validar algoritmos de aprendizaje automático utilizados en sistemas de evaluación automática, con el objetivo de asegurar la precisión y confiabilidad de las decisiones tomadas por el sistema, minimizando la intervención humana.
P5 ¿Cuáles son las etapas de desarrollo de un sistema adaptativo que ajuste su funcionamiento al estudiante, haciendo foco en su desempeño como unidad esencial?	M5. Identificar el proceso y las mejores prácticas para diseñar y desarrollar un sistema de aprendizaje adaptativo que ofrezca una experiencia educativa personalizada y efectiva para cada estudiante

3. Método de revisión

La tarea de localizar la producción científica se inicia mediante la determinación de los parámetros iniciales de búsqueda, los cuales deben estar alineados con la investigación que se está llevando a cabo en este mapeo sistemático de la literatura. Esto permite realizar una primera exploración en el campo. Durante esta etapa, se recopilan las características principales de las diferentes bases de datos, como los parámetros de consulta utilizados para acceder a ellas, el tamaño de la comunidad, la cantidad de citas y el campo disciplinario que abarcan la mayoría de los documentos contenidos en cada base de datos

3.1 Fuentes

Con los parámetros mencionados, se realizar el primer acercamiento a siguientes fuentes: EBSCO, IEEE Explore, Google Académico, Sciencia Direct, y ACM Digital Library. Luego de tomar algunas muestras de documentación en cada base de datos, y analizarlas con foco en las preguntas, las palabras claves y los temas de la investigación, se concluye que las siguientes bases de datos son las más adecuadas para la realización de este mapeo sistemático de la literatura: IEEE Xplore Digital Library; Science Direct, ACM Digital Library

3.2 Definición de términos

El período de búsquedas de este mapeo sistemático de la literatura se produce entre septiembre de 2021 y febrero 2022, e incluye los artículos publicados a partir del año 2012 a la actualidad. El año del parámetro "inicio de búsqueda" no es aleatorio, ya que durante ese año comienzan a aparecer las primeras publicaciones sobre Machine Learning como una disciplina del campo de la Inteligencia Artificial, mediante la cual a través de la identificación de patrones en datos masivos, es posible realizar análisis predictivo.

Tabla 1 Temas principales de búsquedas

TP 1	Ingeniería
TP 2	Inteligencia Artificial
TP 3	Extracción de característica

Tabla 2 Palabras claves – Keywords - (K)

PC 1	Machine learning
PC 2	Artificial intelligence
PC 3	Learning styles.
PC 4	Adaptive systems
PC 5	Smart agents

Las cadenas de búsquedas son utilizadas a los fines de estructurar las palabras y términos claves con operadores adecuados, impulsadas por la investigación. Luego de

la primera entrada al campo, durante el proceso de búsqueda de la localización científica, el análisis de esos primeros documentos determina que, con una sola cadena de búsqueda, era muy posible que se excluyan documentos con contenidos centrales del tema de investigación. La evidencia mostraba que era necesario construir dos cadenas de búsquedas, una orientada a los objetivos generales y otra a los objetivos específicos de la investigación, luego, ya en proceso de escritura podían imbricarse ambas búsquedas y así cubrir todos los aspectos de la investigación. De este proceso teórico, se estructuran dos cadenas de búsquedas (CdB) que se presentan a continuación:

Tabla 3. Cadena de búsqueda

CdB 1	TITLE (“learning styles”) OR TITLE (“adaptive systems with machine learning”) AND TITLE-ABS-KEY (education)
CdB 2	TITLE (“smart agents”) OR TITLE (“adaptive systems”) AND TITLEABS-KEY (“machine learning”)

3.3 Criterios de inclusión y exclusión

Para la selección de este estudio se consideraron los siguientes criterios de inclusión (CI) y de exclusión (CE):

Tabla 4. Criterios de inclusión

CI 1	Publicación científica que responda a las preguntas de investigación AND.
CI 2	Publicación científica que incluye en sus metadatos los temas principales de búsquedas desarrollados por cualquiera de los actores de una comunidad educativa de nivel superior AND.
CI 3	Publicaciones científicas posteriores al año 2012.

Tabla 4. Criterios de exclusión

CE 1	Publicaciones de años anteriores al 2012.
CE 2	Publicaciones alejadas del problema de investigación.
CE 3	Publicaciones que no coincidan con los criterios de inclusión OR
CE 4	Publicaciones repetidas

4. Búsqueda de trabajos

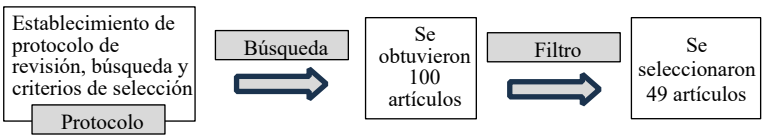
Previa a la creación de la base de datos bibliográficas (BDB), en esta etapa se refina el mapeado de documentos, en la cual se fueron alojando los resultados obtenidos de las cadenas de búsquedas y los alcances de la revisión. A partir de esta entrada al campo, se depuró esta primera masa crítica con el filtro: “Temas principales de la búsqueda”. el resultado fue:

Tabla 5: Temas de búsquedas preliminares

	IEEE	ScDi
Ingeniería	697	55098
Inteligencia artificial	4052	104262
Extracción de características	931	58

Luego de lecturas diagonales en los resúmenes de un gran número de documentos que en sus títulos contenían palabras de las cadenas de búsquedas, se siguió un proceso de tres pasos para obtener los artículos, que se muestran en la Tabla 6. El primer paso consistió en establecer el protocolo de revisión y las herramientas necesarias, lo cual ya se había abordado previamente. Este proceso incluyó la definición de las preguntas de investigación, la creación de la cadena de búsqueda y los criterios de inclusión y exclusión para seleccionar los artículos. También se localizaron mapeos sistemáticos de la literatura relacionados con el tema investigado. Luego se realizó la búsqueda de trabajos en diversas bases de datos, lo que resultó en un total de 100 artículos. Posteriormente, se aplicaron los criterios a esta cantidad de trabajos en un proceso iterativo en el cual se analizaron el título, el resumen y las palabras clave de cada artículo, lo que dejó un total de 49 se utilizaron para este mapeo sistemático.

Tabla 6: Detalle y proceso de búsqueda y filtrado de artículo.



5. Síntesis de datos extraídos

El presente mapeo sistemático de la literatura tuvo como objetivo fortalecer la investigación titulada "Estilos de aprendizaje personalizados con inteligencia artificial". Mediante la búsqueda y selección de trabajos relevantes, se pudo obtener un conjunto de datos que refleja la tendencia predominante en el tema investigado durante el periodo establecido en el protocolo. Gracias a este proceso, se pudo extraer información y identificar las técnicas y tecnologías más utilizadas, con el propósito de responder a las preguntas de investigación planteadas.

P1 ¿Qué estudios primarios existen sobre métodos de enseñanza, modelado con inteligencia artificial considerando los estilos de aprendizaje del estudiante?

Se han realizado diversos estudios sobre este tema, que utilizan herramientas interactivas y modelos basados en taxonomía de Bloom, big data, algoritmos de aprendizaje profundo, minería de datos y ontologías, entre otros enfoques. Algunos de los estudios más relevantes en este campo [12], [14], [19], [23], [28], [32], [36], [40] y [44], en general, demuestran que el modelado con inteligencia artificial de los estilos de

aprendizaje del estudiante puede mejorar significativamente el rendimiento académico y la experiencia de aprendizaje de los estudiantes. Algunos de los enfoques utilizados incluyen la adaptación de la interfaz de usuario [7], [19], [22], [29], la recomendación personalizada de recursos de aprendizaje [4], [7], [12], [13], [19], [26], [27], [29], [33], [35], la mejora de la retroalimentación en tiempo real [13], [19], [24], [29], [27], la implementación de estrategias de metacognición y motivación [33], [32], [19], [40], [42], [45] y la creación de sistemas de aprendizaje adaptativo [7], [9], [612], [19], [27], [29], [31], [33]. Los estilos de aprendizaje mencionados en documentos mapeados son los siguientes [7], [11]:

- Visual: Las personas con este estilo de aprendizaje prefieren aprender a través de imágenes, gráficos y videos. Suelen tener una buena memoria visual y recuerdan mejor lo que ven que lo que escuchan o leen.
- Auditivo: Las personas con este estilo de aprendizaje prefieren aprender a través de la escucha. Suelen tener una buena memoria auditiva y recuerdan mejor lo que escuchan que lo que ven o leen.
- Kinestésico: Las personas con este estilo de aprendizaje prefieren aprender a través de la experiencia práctica y el movimiento. Suelen aprender mejor cuando pueden tocar, mover o experimentar con lo que están aprendiendo.

Por otro lado, surge del análisis que la tecnología educativa, puede ser utilizada de diferentes maneras para adaptarse a los diferentes estilos de aprendizaje. Si se vincula cada estilo mencionado con las posibilidades que ofrece un dispositivo de formación sobre una plataforma digital, podría mencionarse las siguientes características [7], [11]

- Visual: La tecnología educativa puede incluir videos explicativos, gráficos e imágenes para ayudar a los estudiantes con este estilo de aprendizaje a comprender mejor los conceptos.
- Auditivo: La tecnología educativa puede incluir grabaciones de audio, conferencias en línea y podcasts para ayudar a los estudiantes con este estilo de aprendizaje a absorber la información a través de la escucha.
- Kinestésico: La tecnología educativa puede incluir simulaciones, juegos y actividades interactivas para ayudar a los estudiantes con este estilo de aprendizaje a experimentar y aplicar lo que están aprendiendo.

P2 ¿Cuáles métodos y técnicas son las más utilizadas para reconocer los diferentes estilos de aprendizaje de un estudiante?

Los métodos y técnicas más utilizados para reconocer los diferentes estilos de aprendizaje de un estudiante incluyen algoritmos de aprendizaje automático, minería de datos, encuestas, seguimiento ocular y electroencefalografía (EEG) [3], [7], [12], [19], [20], [29], [33]. Un enfoque destacado es el uso de árboles de decisión difusos [7]. Este método permite la clasificación de los estudiantes en diferentes estilos de aprendizaje, incluso con información incierta o ambigua. El seguimiento ocular, combinado con algoritmos de aprendizaje automático [11], también es una técnica relevante. Esta técnica proporciona información valiosa sobre el enfoque del estudiante en la tarea de aprendizaje, mejorando la precisión en la identificación de los estilos de aprendizaje. El

uso de la EEG [20] es otro enfoque prometedor. Esta técnica permite obtener información sobre el procesamiento cognitivo y emocional del estudiante durante la tarea de aprendizaje, lo que facilita la identificación de los estilos de aprendizaje. Los algoritmos de inteligencia computacional también se emplean para reconocer los estilos de aprendizaje [29]. Estos algoritmos ayudan a identificar patrones en los datos de aprendizaje relacionados con los estilos de aprendizaje. El enfoque de la minería de datos y el análisis de los patrones de uso de la web [41] combinan el aprendizaje automático y la minería de datos para detectar automáticamente el perfil del estudiante y sus estilos de aprendizaje, utilizando información sobre los intereses y preferencias de aprendizaje obtenida del análisis de los patrones de uso de la web.

P3 ¿Qué atributos tienen los agentes inteligentes que han sido propuestos como soluciones?

Los atributos de los agentes inteligentes que han sido propuestos como soluciones en el contexto de los estilos de aprendizaje con inteligencia artificial son diversos y se han explorado en varios trabajos de investigación. En [1] proporcionan información general sobre los atributos de los agentes inteligentes, incluyendo la capacidad de aprendizaje automático y adaptación. en [4] presentan un agente inteligente basado en un sistema de tutoría inteligente que utiliza el análisis de conceptos formales para recomendar actividades de aprendizaje adecuadas al estilo de aprendizaje del estudiante. Este agente tiene la capacidad de personalizar las recomendaciones de aprendizaje para satisfacer las necesidades individuales de los estudiantes. El modelado y simulación de un sistema de inferencia neuro-difuso adaptativo (ANFIS) para el aprendizaje móvil son abordados por [9]. En su trabajo describe los atributos de los agentes inteligentes propuestos como soluciones, en particular el uso de técnicas de adaptación y aprendizaje automático para mejorar el rendimiento del agente. En [10] exploran los atributos de los agentes inteligentes propuestos como soluciones en el contexto de las máquinas inteligentes híbridas y destacan la importancia de la capacidad de procesamiento y análisis de datos de los agentes inteligentes. En [15] presenta un sistema de aprendizaje de amplio espectro que utiliza técnicas de transferencia de dominio para la compensación del cambio a largo plazo en sistemas de nariz electrónica y destacan la importancia de la adaptabilidad de los agentes inteligentes para hacer frente a los cambios en el entorno de aprendizaje. Los atributos de los agentes inteligentes se centran en la optimización de los parámetros para mejorar la eficiencia del reconocimiento en un sistema de aprendizaje adaptativo [21]. En [26] se destacan la importancia de la capacidad de aprendizaje profundo de los agentes inteligentes para mejorar la eficacia de los sistemas de recomendación de aprendizaje adaptativo. En [28] menciona que los agentes inteligentes propuestos como soluciones tienen atributos como adaptabilidad, personalización, capacidad de procesamiento y capacidad. Por otro lado, en [26] identifican los atributos clave que deben tener los recommender systems (sistemas de recomendación de aprendizaje), incluyendo la personalización, la adaptabilidad, la transparencia y la retroalimentación. Se identifica la adaptabilidad, la personalización y la retroalimentación como atributos importantes de los agentes inteligentes propuestos como soluciones. Es importante tener en cuenta que no todos estos atributos son igualmente relevantes para cada contexto educativo y que la selección de estos

dependerá de las necesidades específicas de cada caso. Sin embargo, en general, los agentes inteligentes propuestos como soluciones deben ser capaces de adaptarse a las necesidades individuales de los estudiantes y proporcionar retroalimentación y recomendaciones personalizadas basadas en el análisis de datos de aprendizaje

P4 ¿Cuáles son los criterios de validación de un algoritmo de aprendizaje automático de un sistema de evaluación automático, que permitan la toma de decisiones con mínima intervención humana?

Los criterios de validación de un algoritmo de aprendizaje automático en un sistema de evaluación automático que permita la toma de decisiones con mínima intervención humana son diversos y dependen del contexto y los objetivos específicos del sistema. Algunos de los criterios de validación propuestos en la literatura incluyen:

- Precisión de la clasificación: se evalúa la proporción de muestras correctamente clasificadas por el algoritmo, así como las tasas de verdaderos y falsos positivos y negativos [2], [7], [12], [19], [29].
- Capacidad de generalización: se utiliza la validación cruzada y la división del conjunto de datos en entrenamiento y prueba para evaluar la capacidad del modelo de generalizar a nuevos datos [1], [2], [4], [5], [10], [19], [8], [11], [12], [13].
- Tolerancia a fallas y comportamiento erróneo: se evalúa cómo el sistema maneja datos incompletos o ruidosos, así como su comportamiento ante situaciones inesperadas [8].

En cuanto a la validación del sistema de evaluación automático, se propone el uso de experimentos de desarrollo, depuración y verificación del modelo a través de la participación de la comunidad [5], [8], [10], [17],. También se mencionan trabajos específicos que abordan la validación de algoritmos de aprendizaje automático en diferentes contextos, como la identificación del lugar de origen de los estudiantes y el desarrollo de productos [1], [4], [7], [9], [10], [12], [13], [19].

P5 ¿Cuáles son las etapas de desarrollo de un sistema adaptativo que ajuste su funcionamiento al estudiante, haciendo foco en su desempeño como unidad esencial?

Las etapas de desarrollo de un sistema adaptativo que ajuste su funcionamiento al estudiante, haciendo foco en su desempeño como unidad esencial, se pueden resumir en cinco etapas principales:

- Recopilación y análisis de datos de aprendizaje: Se recopilan datos relevantes sobre el desempeño y las preferencias del estudiante, así como información contextual relacionada. Estos datos se analizan para identificar patrones y características clave [16], [19], [22], [24], [25], [27], [29].
- Creación de un modelo de usuario: Se utiliza la información recopilada para construir un modelo del estudiante que capture sus características, estilos de aprendizaje y necesidades individuales. Este modelo sirve como base para la adaptación del sistema [7], [12], [22], [29], [9].
- Selección de estrategias de adaptación: Se eligen estrategias de adaptación

apropiadas basadas en el modelo de usuario y en el conocimiento pedagógico. Estas estrategias determinan cómo se ajustará el sistema para satisfacer las necesidades del estudiante [7], [19].

- Implementación de la estrategia de adaptación elegida: Se realiza la implementación técnica de las estrategias de adaptación seleccionadas. Esto implica desarrollar y configurar los componentes del sistema que permitirán ajustar la presentación del contenido, la retroalimentación y las actividades de acuerdo con las características del estudiante [4], [7], [9], [12], [13], [14], [22], [24], [25], [30], [31], [33], [35].
- Evaluación del sistema: Se evalúa el desempeño del sistema adaptativo en función de su capacidad para mejorar el aprendizaje y el rendimiento del estudiante. Se realizan pruebas y análisis para medir la efectividad de las estrategias de adaptación y se realizan ajustes y mejoras según sea necesario [4], [7], [12], [19], [22], [27], [28], [29], [31].

Estos pasos se han abordado en los estudios citados como el marco de trabajo para el aprendizaje activo en la educación de ingeniería, la selección de estrategias de adaptación basada en el análisis del comportamiento de aprendizaje en línea, y la implementación de sistemas adaptativos basados en inteligencia artificial. La literatura existente proporciona una variedad de enfoques y técnicas para cada etapa del desarrollo de un sistema adaptativo [7]. [9]. [13].

Tabla 7: Principal aporte de cada artículo.

Preguntas de investigación	Aportes más relevantes
P1 ¿Qué estudios primarios existen sobre métodos de enseñanza, modelado con inteligencia artificial considerando los estilos de aprendizaje del estudiante?	[4], [6], [12], [13], [14], [19], [23], [28], [32], [35], [36], [40], [44], [46], [47], [49].
P2 ¿Cuáles métodos y técnicas son las más utilizadas para reconocer los diferentes estilos de aprendizaje de un estudiante?	[7], [11], [20], [24], [29], [33], [41], [46], [47], [49].
P3 ¿Qué atributos tienen los agentes inteligentes que han sido propuestos como soluciones?	[1], [3], [4], [8], [9], [10], [15], [21], [26], [28], [34], [38].
P4 ¿Cuáles son los criterios de validación de un algoritmo de aprendizaje automático de un sistema de evaluación automático, que permitan la toma de decisiones con mínima intervención humana?	[1], [5], [8], [10], [15], [18], [25], [31], [37], [42].
P5 ¿Cuáles son las etapas de desarrollo de un sistema adaptativo que ajuste su funcionamiento al estudiante, haciendo foco en su desempeño como unidad esencial?	[3], [6], [12], [17], [22], [27], [30], [35], [39], [43], [45], [48]

6. Conclusiones

El mapeo sistemático de la literatura ha permitido identificar una serie de estudios y enfoques que demuestran la efectividad de la utilización de inteligencia artificial para adaptar las estrategias de enseñanza a los estilos de aprendizaje de los estudiantes. Algunos enfoques utilizados incluyen la adaptación de la interfaz de usuario, la recomendación personalizada de recursos de aprendizaje, la mejora de la retroalimentación en tiempo real y la implementación de estrategias de metacognición y motivación. Se identificaron tres estilos de aprendizaje principales: visual, auditivo y kinestésico, y se propusieron diversas formas en las que la tecnología educativa puede adaptarse a cada estilo. Estos estudios indican que la personalización de la enseñanza basada en inteligencia artificial puede mejorar significativamente el rendimiento académico y la experiencia de aprendizaje de los estudiantes.

Referencias

[1] J. Singh, G. Dhiman, "A survey on machine-learning approaches: Theory and their concepts," *Materials Today: Proceedings*, 2021.

[2] J. Bas, C. Cirillo, E. Cherchi, "Classification of potential electric vehicle purchasers: A machine learning approach," *Technological Forecasting and Social Change*, vol. 168, 2021.

[3] F. Maffei, C. Neil, N. Battaglia, "Herramientas para determinar estilos de aprendizaje basadas en Inteligencia artificial," in *XVII Congreso de Tecnología en Educación & Educación en Tecnología - TE&ET,* 2022.

[4] J. Muangprathub, V. Boonjing, K. Chamnongthai, "Learning recommendation with formal concept analysis for intelligent tutoring system," *Heliyon*, vol. 6, issue 10, 2020.

[5] K. Wang, J. J. Lu, "A Two-Layer Risky Driver Recognition Model With Context Awareness," *IEEE Access*, vol. 9, pp. 138483-138495, 2021.

[6] K. C. Chilukuri, "A Novel Framework for Active Learning in Engineering Education Mapped to Course Outcomes," *Procedia Computer Science*, vol. 172, pp. 28-33, 2020.

[7] K. Crockett, A. Latham, N. Whitton, "On predicting learning styles in conversational intelligent tutoring systems using fuzzy decision trees," *International Journal of HumanComputer Studies*, vol. 97, pp. 98-115, 2017.

[8] L. Myllyaho, M. Raatikainen, T. Männistö, J. K. Nurminen, T. Mikkonen, "On misbehaviour and fault tolerance in machine learning systems," *Journal of Systems and Software*, vol. 183, p. 111096, 2022.

[9] J. Al-Hmouz, R. Shen, R. Al-Hmouz, J. Yan, "Modeling and Simulation of an Adaptive Neuro-Fuzzy Inference System (ANFIS) for Mobile Learning," *IEEE Transactions on Learning Technologies*, vol. 5, no. 3, pp. 226-237, 2012.

[10] B. S., N. Soni, and S. Dixit, "Crowdsourcing – A Step Towards Advanced Machine Learning," *Procedia Computer Science*, vol. 132, pp. 632-642, 2018.

[11] F. Klaib, N. O. Alsrehin, W. Y. Melhem, H. O. Bashtawi, and A. A. Magableh, "Eye tracking algorithms, techniques, tools, and applications with an emphasis on machine learning and Internet of Things technologies," *Expert Systems with Applications*, vol. 166, p. 114037, 2021.

[12] Viloria, I. R. P. Gonzalez, and O. B. P. Lezama, "Learning Style Preferences of College Students Using Big Data," *Procedia Computer Science*, vol. 160, pp. 461-466, 2019.

[13] B. F. Mansur, N. Yusof, and A. H. Basori, "Personalized Learning Model based on Deep Learning Algorithm for Student Behaviour Analytic," *Procedia Computer Science*, vol. 163, pp. 125-133, 2019.

[14] Jayasingh, "A Data Mining Approach to Inquiry Based Inductive Learning Practice in Engineering Education," in *2016 IEEE 6th International Conference on Advanced Computing (IACC)*, Bhimavaram, pp. 845-850. India, 2016,

[15] X. Liu, F. Zeng, S. Tian, S. Zhang, and L. Zhao, "Domain Transfer Broad Learning System for Long-Term Drift Compensation in Electronic Nose Systems," in *IEEE Access*, vol. 7, pp. 143947-143959, 2019.

[16] V. Verma, Z. Stoffová, S. Illés, S. Tanwar, and N. Kumar, "Machine Learning-Based Student's Native Place Identification for Real-Time," in *IEEE Access*, vol. 8, pp. 130840130854, 2020.

[17] Deng, X. Ji, C. Rainey, J. Zhang, and W. Lu, "Integrating Machine Learning with Human Knowledge," *iScience*, vol. 23, no. 11, p. 101656, 2020.

[18] L. K. Yamamura, C. O. Ribeiro, D. Dantas, J. A. Quintanilha, and F. T. Berssaneti, "The Front-End of Product Development as Systems Thinking and Predictive Learning," *Procedia Manufacturing*, vol. 39, pp. 1346-1353, 2019.

[19] Lwande, L. Muchemi, and R. Oboko, "Identifying learning styles and cognitive traits in a learning management system," *Heliyon*, vol. 7, no. 8, p. e07701, 2021.

[20] Deenadayalan, A. Kangaiammal, and B. K. Poornima, "EEG Based Learner's Learning Style and Preference Prediction for E-learning," in *2018 2nd International Conference on ISMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, pp. 316-320, 2018.

[21] D.-P. Tran, G.-N. Nguyen, and V.-D. Hoang, "Hyperparameter Optimization for Improving Recognition Efficiency of an Adaptive Learning System," in *IEEE Access*, vol. 8, pp. 160569160580, 2020.

[22] Díaz, C. Cubillos, R. Mellado, and E. Barbaguelatta, "Development of a prototype of elearning based on ontologies to analyze the impact of learning styles on engineering students," in *2017 36th International Conference of the Chilean Computer Science Society (SCCC)*, pp. 1-9, 2017.

[23] V. Kowalski and S. E. Kowalski, "The effect of student learning styles on the learning gains achieved when interactive simulations are coupled with real-time formative assessment via pen-enabled mobile technology," in *2012 Frontiers in Education Conference Proceedings*, Seattle, WA, USA, pp. 1-5, 2012.

[24] J. Painén-Aravena, J. Alfaro-Pérez, P. Ramírez-Correa, E. E. Grandón and S. ArayaGuzmán, "Investigating the effect of learning styles on the acceptance of e-books among university students," in *2019 14th Iberian Conference on Information Systems and Technologies (CISTI)*, Coimbra, Portugal, pp. 1-4, 2019.

[25] G. T. Huamani and P. M. T. Inga, "WIP Adaptive evaluation for a systems theory course according to the learning context," in *2021 IEEE World Conference on Engineering Education (EDUNINE)*, Guatemala City, Guatemala, pp. 1-4, 2021.

[26] Gomede, R. Miranda de Barros, and L. de Souza Mendes, "Deep auto encoders to adaptive E-learning recommender system," *Computers and Education: Artificial Intelligence*, vol. 2, 100009, 2021.

[27] H. Chen, C. Yin, R. Li, W. Rong, Z. Xiong, and B. David, "Enhanced learning resource recommendation based on online learning style model," *Tsinghua Science and Technology*, vol. 25, no. 3, pp. 348-356, 2020.

[28] H. A. El-Ghareeb, "Intelligent and adaptive microservices and neutrosophic-based learning management systems," in *Optimization Theory Based on Neutrosophic and Plithogenic Sets*, F. Smarandache and M. Abdel-Basset, Eds. Academic Press, pp. 63-85,2020.

[29] J. Bernard, T.-W. Chang, E. Popescu, and S. Graf, "Learning style Identifier: Improving the precision of learning style identification through computational intelligence algorithms," *Expert Systems with Applications*, vol. 75, pp. 94-108, 2017.

[30] M. Ciolacu, A. F. Tehrani, L. Binder and P. M. Svasta, "Education 4.0 - Artificial Intelligence Assisted Higher Education: Early recognition System with Machine Learning to support Students' Success," in *2018 IEEE 24th International Symposium for Design and Technology in Electronic Packaging (SIITME)*, Iasi, Romania, pp. 23-30, 2018

[31] M. Dlamini and W. S. Leung, "Evaluating Machine Learning Techniques for Improved Adaptive Pedagogy," in *2018 IST-Africa Week Conference (IST-Africa)*, Gaborone, Botswana, Page 10 of 10, 2018.

[32] M. H. Miller, J. DeClerck, W. J. Endres, L. Roberts, K. Hale and S. Sorby, "Evaluation of computer modules to teach metacognition and motivation strategies," in *2013 IEEE Frontiers in Education Conference (FIE)*, Oklahoma City, OK, USA, pp. 707-709, 2013.

[33] M. Hariharan, K. Sooda, N. Vineeth and G. S. Rekha, "Teaching Style Recommender using Machine Learning," in *2019 1st International Conference on Advances in Information Technology (ICAIT)*, Chikmagalur, India, pp. 238-244, 2019.

[34] M. Hasanuzzaman Shawon, S. M. Muyeen, A. Ghosh, S. M. Islam and M. S. Baptista, "Multi-Agent Systems in ICT Enabled Smart Grid: A Status Update on Technology Framework and Applications," in *IEEE Access*, vol. 7, pp. 97959-97973, 2019.

[35] M. Kaushik and G. Joshi, "Transitional Learning Style Preferences and Its Factors in Newer Generation Engineering Students," in *2016 IEEE 4th International Conference on MOOCs, Innovation and Technology in Education (MITE)*, Madurai, India, pp. 263-267, 2016.

[36] M. M. Mohamad, A. R. Jamali, M. I. Mukhtar, L. C. Sern and A. Ahmad, "Learning styles and critical thinking skills of engineering students," in *2017 IEEE 9th International Conference on Engineering Education (ICEED)*, Kanazawa, Japan, pp. 54-58, 2017.

[37] M. T, A. S and K. K, "Using item response theory in machine learning algorithms for student response data," in *2021 IEEE International Conference on Smart Information Systems and Technologies (SIST)*, Nur-Sultan, Kazakhstan, pp. 1-5, 2021.

[38] Maha Faisal, Aisha Bourahma, Feda AlShahwan,. "Towards a reference model for sensorsupported learning systems," *Journal of King Saud University - Computer and Information Sciences*, vol. 33, no. 9, pp. 1145-1157, 2021.

[39] Michał Choraś, Konstantinos Demestichas, Agata Giełczyk, Álvaro Herrero, Paweł Ksieniewicz, Konstantina Remoundou, Daniel Urda, Michał Woźniak, "Advanced Machine Learning techniques for fake news (online disinformation) detection: A systematic mapping study," *Applied Soft Computing*, vol. 101, 2021.

[40] Aissaoui, O.E. & Oughdir, L. "A learning style-based Ontology Matching to enhance learning resources recommendation." In *1st International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)*, Meknes, Morocco, pp. 1-7. 2020.

[41] El Aissaoui, O., El Madani El Alami, Y., Oughdir, L., & El Allioui, Y. "Integrating web usage mining for an automatic learner profile detection: A learning styles-based approach." In *International Conference on Intelligent Systems and Computer Vision (ISCV)*, Fez, Morocco, pp. 1-6, 2018.

[42] P. K. Tulsi, M. P. Poonia and Anupriya, "Learning styles and achievement of engineering students," *2016 IEEE Global Engineering Education Conference (EDUCON)*, Abu Dhabi, United Arab Emirates, pp. 192-196, 2016.

[43] P. Poulova and I. Simonova, "Individual learning styles and university students," *Proceedings of the 2012 IEEE Global Engineering Education Conference (EDUCON)*, Marrakech, Morocco, pp. 1-6, 2012.

[44] P. Tsai and H. Liao, "Students' progressive behavioral learning patterns in using machine translation systems – A structural equation modeling analysis," *System*, vol. 101, 2021.

[45] P. Różewski, M. Kieruzel, T. Lipczyński, M. Prys, M.-A. Sicilia, E. García-Barriocanal, S.

Sánchez-Alonso, C. Hamill, C. Royo, and F. Uras, "Concept of expert system for creation of personalized, digital skills learning pathway," *Procedia Computer Science*, vol. 159, pp. 23042312, 2019.

[46] R. Bajaj and V. Sharma, "Smart Education with artificial intelligence based determination of learning styles," *Procedia Computer Science*, vol. 132, pp. 834-842, 2018.

[47] S. V. Kolekar, R. M. Pai, and M. Pai M.M., "Adaptive User Interface for Moodle based Elearning System using Learning Styles," *Procedia Computer Science*, vol. 135, pp. 606-615, 2018.

[48] Tumaini Kabudi, Ilias Pappas, Dag Håkon Olsen, "AI-enabled adaptive learning systems: A systematic mapping of the literature," *Computers and Education: Artificial Intelligence*, vol. 2, 2021.

[49] Yu Hsin Hung, Ray I. Chang, Chun Fu Lin, "Hybrid learning style identification and developing adaptive problem-solving learning activities," *Computers in Human Behavior*, vol. 55, Part A, pp. 552-561, 2016.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Desarrollo de competencias conducido por rúbricas

Carlos Neil, Nicolás Battaglia, Marcelo De Vincenzi

Universidad Abierta Interamericana. Facultad de Tecnología Informática.
Centro de Altos Estudios en Tecnología Informática. Buenos Aires. Argentina
{carlos.neil,nicolas.battaglia,medevincenzi}@uai.edu.ar

Resumen. Se presenta un modelo para el desarrollo de competencias que utiliza conceptos del aprendizaje basado en problemas como estrategia de aprendizaje y descriptores de la rúbrica, asociados al problema, como criterios de evaluación. La propuesta se enmarca en el alineamiento constructivo que valora la coherencia entre los resultados de aprendizaje, su desarrollo y evaluación y se fundamenta en la teoría de la carga cognitiva que postula que el esfuerzo impuesto a los estudiantes durante una tarea de aprendizaje influye en su capacidad para adquirir y transferir conocimientos. El proceso vincula, en una secuencia didáctica, cada criterio de evaluación de una rúbrica con trabajos prácticos de baja complejidad cuya solución deje en evidencia el criterio valorativo establecido. Esta estrategia permite que el estudiante se concentre, inicialmente, en aspectos puntuales, para luego integrarlos en un trabajo práctico que utiliza como criterio de evaluación la rúbrica asociada.

Palabras clave: alineamiento constructivo, carga cognitiva, resultados de aprendizaje, rúbricas, secuencia didáctica.

1. Introducción

El alineamiento constructivo [1–3] es un enfoque pedagógico que resalta la importancia de la coherencia entre los resultados de aprendizaje, el proceso de aprendizaje y la evaluación. Se orienta a garantizar que las estrategias de aprendizaje y evaluación implementadas en los programas de estudio estén en concordancia con los resultados de aprendizaje. Es necesario que las actividades pedagógicas estén diseñadas de manera coherente con los resultados de aprendizaje y los criterios de evaluación. Además, los métodos y criterios de valorativos, que deben medir de manera efectiva el logro de los resultados de aprendizaje, deben ser claros y coherentes con los estándares de desempeño esperados.

Se denomina *micro TP* a una actividad de complejidad mediana o baja que se enfoca a una temática particular y cuya resolución puede completarse en un lapso de entre 30 a 45 minutos. Los micro TP se complementan con las rúbricas como guías que dirigen el proceso de aprendizaje mediante la vinculación de los criterios de evaluación (dimensiones en los que descompone el problema) con actividades simples que dividen el problema en partes (análisis) en pos de la posterior integración (síntesis) de sus componentes. El uso de criterios de evaluación de rúbricas como guía para la resolución de los micro TP proporciona una estructura clara que orienta el aprendizaje y permite comprender mejor los componentes esenciales del problema y cómo se relacionan entre sí una vez integrados. Al establecer una partición del problema, los estudiantes abordan cada una de ellas de forma más efectiva, lo que facilita la síntesis final y la comprensión integral del problema. En resumen, el uso combinado de criterios de evaluación para la

resolución de micros TP no solo mejora la evaluación del aprendizaje, sino que también facilita el proceso disminuyendo la carga de trabajo de los estudiantes.

De los tres componentes del alineamiento constructivo, en trabajos previos se han desarrollado dos propuestas: en [4], se presentó un proceso para el diseño de rúbricas y un repositorio abierto a la comunidad educativa y, en [5], los lineamientos para la escritura de resultados de aprendizaje. Para completar la trilogía, se presenta un modelo de desarrollo de competencias conducido por rúbricas que utiliza conceptos del aprendizaje basado en problemas [6–9] como estrategia de aprendizaje y descriptores de la rúbrica como criterios de evaluación.

Este artículo forma parte del proyecto de investigación *Herramientas Colaborativas Multiplataforma en la Enseñanza de la Ingeniería de Software*, desarrollado en el Centro de Altos Estudios en Tecnología Informática [10] de la Universidad Abierta Interamericana.

2. Componentes del modelo

Se detallan los principales conceptos, herramientas y teorías utilizados en el modelo. Primero, se describen a los resultados de aprendizaje como direccionadores del proceso; luego, a las rúbricas como estrategia de evaluación (y aprendizaje); a continuación, se exponen conceptos de la teoría de la carga cognitiva que fundamenta el uso de micro TP y que, posteriormente, se detallan en sus características principales y, por último, se describe a la secuencia didáctica como integradora de todo el proceso.

2.1 ¿Cómo se describen las competencias? Resultados de aprendizaje

El modelo educativo basado en competencias y centrado en el estudiante destaca a los resultados de aprendizaje como clave para guiar el proceso de aprendizaje. Se necesitan lineamientos claros para estandarizar una escritura que permita ser comprendida tanto por la comunidad educativa como la externa. Comenzar con la matriz de competencias para identificar los resultados de aprendizaje relacionados con las asignaturas y los niveles de dominio asociados permite, por un lado, asegurar la coherencia entre competencias de egreso y resultados de aprendizaje y, por otro, establecer lineamientos precisos para obtener claridad en su descripción [5].

Para evaluar el desarrollo de competencias se utilizan niveles de dominio que determinan el progreso en el aprendizaje a lo largo del tiempo y que permiten identificar áreas donde reforzar algún aspecto [11]. La matriz de competencias describe cómo cada asignatura contribuye al desarrollo de una o más competencias, considerando niveles de complejidad, integración y autonomía esperados del estudiante. Cada asignatura se enfoca en promover un conjunto limitado de resultados de aprendizaje, lo que permite el desarrollo gradual y planificado de las competencias de egreso a lo largo del recorrido académico [12].

En resumen, la escritura de resultados de aprendizaje recorre un proceso iterativo e incremental que comienza identificando a las competencias de egreso y los niveles de dominio asociados a la asignatura. Luego, para su escritura, se determina el verbo según el nivel de dominio establecido, el objeto de conocimiento, la finalidad y, por último,

la condición de referencia. A partir de este proceso se desarrollan, para cada asignatura del plan de estudios, el conjunto de resultados de aprendizaje que ese espacio curricular se compromete a desarrollar [5].

2.2 ¿Cómo se evalúan a las competencias? Rúbricas analíticas

La evaluación orienta y motiva el aprendizaje y la propia enseñanza [13] y no debe ser considerada como un proceso escindido del aprendizaje, sino una oportunidad para fomentarlo [14]. Las rúbricas no se limitan a la evaluación, también pueden utilizarse en estrategias de aprendizaje al descomponer tareas complejas, relacionadas con competencias, en tareas más simples, distribuidas de forma gradual y operativa [15]. El uso de rúbricas se fortalece en los entornos virtuales de enseñanza y aprendizaje, ya que su diseño y utilización empoderan al docente, al estudiante y a la institución, fomentando así una cultura de evaluación [16]. Las rúbricas son un instrumento válido para promover el aprendizaje, especialmente cuando se combinan con diferentes actividades metacognitivas, como la autorregulación, la coevaluación y la autoevaluación [17].

La rúbrica presentada en [4] evalúa el nivel de logro en la resolución de problemas al relacionar indicadores con niveles de dominio que incluyen un formato de descriptores analíticos que desambigua su uso. El rasgo diferenciador está centrado en la construcción de los *descriptores*; cada criterio de evaluación se puede subdividir en uno o más descriptores y, cada uno de ellos, a su vez, se compone de un *contexto* vinculado y dos *criterios analíticos A+ y B*, que podrán ser contrastados con el trabajo evaluado para establecer si lo cumplen, o no; ver figura 2 en [4].

Los descriptores de las rúbricas serán utilizados, en el proceso de desarrollo de competencias, como criterios de evaluación en la resolución de los micro TP y la rúbrica completa para evaluar el TP integrador.

2.3 ¿Qué características deben tener los trabajos prácticos? Carga cognitiva

Los micros TP se fundamentan en la teoría de la carga cognitiva [18] que refiere al impacto de la carga mental en el proceso de solución de problemas y en el aprendizaje. Una carga cognitiva alta puede dificultar el aprendizaje ya que consume más recursos mentales y limita la capacidad de procesamiento. El esfuerzo del estudiante se puede reducir mediante estrategias como la simplificación del problema, la presentación de información relevante y la guía del proceso de resolución. Instrucciones claras y retroalimentación oportuna, ayudan a optimizar la carga de trabajo. La reducción de la carga cognitiva implica simplificar la información y los procedimientos para que sean más fáciles de procesar. Presentar la información de manera estructurada y secuencial facilita su comprensión y proporcionar ejercicios y casos prácticos con instrucciones claras ayuda a los estudiantes a aplicar los conocimientos adquiridos. Una retroalimentación oportuna permite a los estudiantes corregir errores y mejorar su desempeño. Distribuir el aprendizaje en el tiempo, con intervalos de descanso entre las sesiones de estudio, favorece la retención a largo plazo. Por otro lado, diseñar situaciones de aprendizaje que reflejen los desafíos del entorno laboral real, facilitan la transferencia de conocimientos [19]. Sumado a esto, la secuenciación de las actividades

de aprendizaje según la complejidad y familiaridad de los conceptos optimiza la carga cognitiva y permite a los estudiantes desarrollar su capacidad para manejarla, mejorando así su rendimiento. El seguimiento en tiempo real del desempeño del estudiante mientras interactúa con el sistema de aprendizaje y la retroalimentación inmediata permite corregir errores y fortalecer áreas débiles [20].

2.4 ¿Cómo se estructuran los trabajos prácticos? Micro TP

Los *micro TP* son actividades de complejidad mediana o baja que se enfocan a una temática particular y cuya resolución puede completarse en un lapso de entre 30 a 45 minutos. Para su diseño se utilizan conceptos del Aprendizaje Basado en Problemas (ABP) definido como un conjunto de actividades que fomentan el pensamiento complejo y la reflexión grupal y cooperativa para tomar decisiones sobre problemas reales y relevantes dentro del ámbito profesional en el que participa y se forma el estudiante [21]. Se centra en el desarrollo de competencias a través de la resolución de problemas en donde los estudiantes se enfrentan, en equipos colaborativos, a situaciones problemáticas del mundo real para identificar y resolverlos, utilizando el conocimiento y las habilidades adquiridas.

Los beneficios del ABP se pueden detallar a partir de sus características [9]. Por ejemplo, estimula la creatividad al fomentar la generación de ideas y el desarrollo del pensamiento crítico; promueve la organización de ideas y potencia el análisis, síntesis y evaluación, al enfrentarse a problemas complejos; genera mayor motivación al trabajar con problemas del mundo real; promueve el descubrimiento autónomo de conocimientos ya que en lugar de recibir información de manera pasiva, los estudiantes participan activamente en su búsqueda y construcción, lo que fomenta su autonomía y capacidad para aprender de manera independiente; incentiva la participación y desarrolla destrezas para el trabajo colaborativo y la resolución de problemas; fomenta la comunicación interpersonal, el trabajo en equipos multidisciplinarios y la mejora a través de la retroalimentación; desarrolla habilidades en relaciones interpersonales, al trabajar en equipos así como habilidades sociales, como la empatía, la escucha activa y la negociación, que son fundamentales en las interacciones interpersonales.

Un aspecto importante es el diseño del TP integrador. En [5] se utilizó la taxonomía de Bloom [22] para definir los niveles de dominio y la elección del verbo en los resultados de aprendizaje. Se recomienda el uso de la taxonomía para definir las características del problema que, en este caso, deberán corresponderse con los niveles cognitivos superiores: análisis, síntesis y evaluación [23]. Esto implica proponer problemas de la vida real para que el estudiante se involucre con ellos y en donde la información necesaria para resolverlos esté incompleta, para se vean obligados a investigar, descubrir nuevos materiales de estudio, hacer juicios y tomar decisiones. Por último, la solución del problema deberá tener más de una respuesta aceptable en función de las premisas establecidas ya que se busca que los estudiantes exploren diversas soluciones o enfoques para resolverlos. Esto estimula la creatividad y el pensamiento innovador.

2.5 ¿Cómo se implementa el proceso? Secuencia didáctica

El desarrollo de competencias requiere una secuencia didáctica [11, 24, 25] entendida como el conjunto articulado de actividades de aprendizaje y evaluación que, con la mediación docente, persigan el logro de determinadas metas educativas, considerando una serie de recursos. Se identifican, a continuación, los componentes de una secuencia didáctica por competencias [11] y se establecen correspondencias con el modelo propuesto. En primer lugar, se define la *Situación problema de contexto* que pretende desarrollar a la competencia, en el modelo presentado, los micro TP y el TP integrador; luego, las *Competencias a formar* que están representados por los resultados de aprendizaje; continúa con las *Actividades de aprendizaje y evaluación*, que indican las actividades con el docente y las actividades de aprendizaje autónomo de los estudiantes, en la propuesta, el monitoreo y las retroalimentación del docente y el desarrollo de los micro TP y el TP integrador de los estudiantes; luego la *Evaluación*, donde se establecen los criterios y evidencias para orientar la evaluación del aprendizaje y la ponderación respectiva, en el proceso, se corresponden con los descriptores que evalúan a los micro TP y con la rúbrica que valora el TP integrador; se continúa con los *Recursos*, es decir, los materiales educativos requeridos que, en el modelo, es la aplicación UAIRubric [26]; por último, el *Proceso metacognitivo*, donde se describen las principales sugerencias para que el estudiante reflexione y se autorregule en el proceso de aprendizaje, en el modelo propuesto, la auto y coevaluación y la retroalimentación del docente (heteroevaluación).

3. Proceso de desarrollo de competencias

Durante el desarrollo de competencias es esencial monitorear y ajustar la carga de trabajo para garantizar un aprendizaje óptimo que permita adaptar las estrategias de enseñanza, proporcionar retroalimentación adecuada y ofrecer oportunidades de práctica adicional según las necesidades individuales de los estudiantes [27]. Aplicar la teoría de la carga cognitiva en el desarrollo de competencias permite diseñar estrategias de enseñanza y evaluación más efectivas que ayuden a los estudiantes a desarrollar las habilidades necesarias para abordar problemas de manera más eficiente y efectiva. Dividiendo una tarea compleja en pasos más simples y manejables se reduce el esfuerzo global, facilitando el aprendizaje. Al final, con un trabajo práctico integrador, se produce la síntesis de todo el proceso.

3.1 Modelo del proceso de desarrollo de competencias

El modelo de desarrollo de competencias involucra a la matriz de competencias, los resultados de aprendizaje, los criterios de evaluación de la rúbrica y los micro TP que se complementan con un TP integrador (Fig. 1):

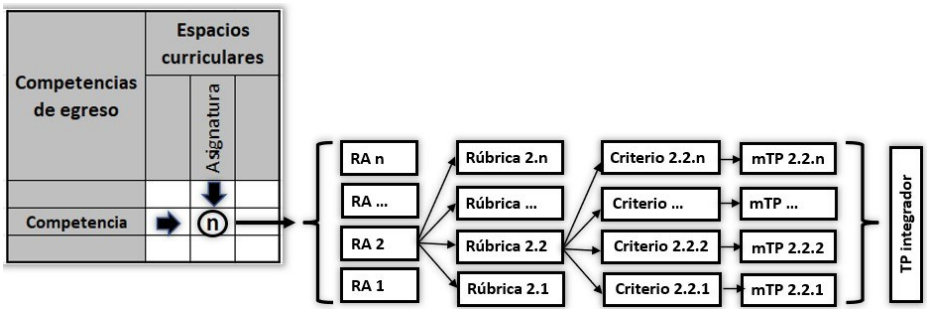


Fig. 1. Modelo de desarrollo de competencias conducido por rúbricas

Se identifica en la matriz de competencias la *Asignatura* y el nivel de dominio asociado; esto implica que esa *Asignatura* contribuirá con un conjunto de resultados de aprendizaje cuyo objetivo será desarrollar la *Competencia* de egreso establecida en la matriz [5]. La *Asignatura* está asociada a un conjunto de resultados de aprendizaje (*RA 1,..., n*); en particular, el *RA 2*, se vincula con un conjunto de rúbricas (*Rúbrica 2.1,..., 2.n*); una de ellas, por ejemplo, la *Rúbrica 2.2,* se divide en una serie de criterios de evaluación (*Criterios 2.2.1,..., 2.2.n*) que tienen asignados, respectivamente, micros TP (*mTP 2.2.1,...,2.2.n)*. Por último, todas las temáticas desarrolladas en los micro TP se retoman y vinculan en un *TP integrador*.

En resumen, el proceso relaciona cada descriptor de la rúbrica con micros TP cuya solución deje en evidencia los criterios de evaluación establecidos y donde el estudiante solo se concentre en los aspectos considerados, lo que permite disminuir la carga cognitiva al concentrar su atención en un tema específico. Los micro TP asociados se resuelven a partir de una secuencia didáctica que finaliza con un TP integrador que se evalúa con la rúbrica completa.

3.2 Desarrollo de competencias conducido por rúbricas: un caso práctico

Para ejemplificar el proceso se utiliza la rúbrica de diagrama de clases UML del *Repositorio de rúbricas analíticas* [26]. En sus metadatos se detalla la asignatura asociada, el resultado de aprendizaje y nivel de dominio. En la rúbrica se identifican los criterios de evaluación con sus descriptores y criterios A+ y B asociados que, por falta de espacio, invitamos al lector a verlos en [26].

La actividad se desarrolla en la asignatura cuatrimestral *Análisis y diseño de sistemas II* del segundo año de la carrera de ingeniería en sistemas informáticos de la Universidad Abierta Interamericana, con una carga de 5 horas semanales. Tributa a la competencia de egreso *Especificar, proyectar y desarrollar sistemas de información*, con un nivel de dominio "2"; esto implica que las actividades de aprendizaje estarán enfocadas en el desarrollo de habilidades y procedimientos para la resolución de problemas relevantes en situaciones contextuales estructuradas con un grado relativo de autonomía. La actividad, alineada al resultado de aprendizaje establecido, tiene asignada dos semanas y corresponde a la unidad II *Herramientas de modelado del análisis y diseño orientado a objetos* del programa de la asignatura que se desarrolla con actividades sincrónicas y asincrónicas autogestivas.

La asignatura también aporta a la competencia transversal *Competencia para desempeñarse de manera efectiva en equipos de trabajo* con un nivel de dominio "2". Por esta razón, las actividades de aprendizaje se desarrollarán de manera colaborativa en equipos de 3 o 4 estudiantes, utilizando conceptos del APB. Los criterios de evaluación para el trabajo en equipo se utilizarán, además, como lineamientos generales para el desarrollo de esa competencia, éstos se detallan en la rúbrica *Trabajo en grupo* del repositorio [26].

Se presenta una secuencia didáctica como opción de actividad áulica que puede utilizarse como modelo para otras alternativas.

Previo a la clase presencial, se propone a los estudiantes una lectura vinculada con el modelo UML [28] para que comprendan la problemática de los estándares. Además, se les solicita que lean el contenido que está en la plataforma de educación a distancia para interiorizarse en conceptos teóricos y prácticos de la temática a desarrollar (aula invertida). Se complementa con un cuestionario que permite detectar problemas en la comprensión de los saberes.

En la primera clase sincrónica, el docente propone la resolución, en equipos, de los *micros TP 1, 2 y 3* vinculados a los criterios de evaluación *CE 1, 2 y 3*, respectivamente. La estrategia de aula invertida permite al docente centrar su labor áulica en actividades de tutorías, por lo tanto, en la clase sincrónica, cumple ese rol respondiendo a las consultas de los estudiantes. Cada micro TP finalizado es entregado al docente que los evalúa utilizando los criterios establecidos en los descriptores, agregando, además, comentarios y sugerencias, lo que podría implicar, en caso de ser necesario, una nueva entrega. Posteriormente, en modalidad asincrónica, los estudiantes desarrollan, grupalmente, los *micro TP 4, 5 y 6* utilizando los criterios de evaluación *CE 4, 5 y 6,* respectivamente; finalizados, cada equipo los entrega al docente para su evaluación. Para finalizar, el docente evalúa y entrega cada uno de los micro TP que incluye comentarios y sugerencias que, en caso de ser necesario, la volverán enviar si es que el docente así lo determina.

En la segunda semana, en la actividad sincrónica se desarrolla, en equipos, el *TP integrador* que utiliza como criterios de evaluación a la rúbrica completa. Finalizado el trabajo en equipos, el docente presenta una versión del TP integrador con una posible solución (no la única) del problema planteado. Como actividad asincrónica, se solicita a cada grupo que desarrolle la versión final del TP integrador, que se autoevalúe y coevalúe el TP integrador de otro equipo y, luego, los envíe al docente. Por último, con el fin de desarrollar actividades metacognitivas, cada estudiante autoevalúa su actividad colaborativa utilizando la rúbrica *Trabajo en grupo* del repositorio [26].

Para finalizar, una vez completadas las actividades de auto y coevaluación, el docente realiza la heteroevaluación enviando comentarios y sugerencias que podrían implicar la necesidad de una nueva versión del TP integrador. Este proceso continúa hasta que el docente considere que los niveles de logro de la competencia estén alineados al nivel de dominio establecido.

3.3 Consideraciones finales

Considerando el tiempo que el docente dedica para las devoluciones de los micro TP y del TP integrador, es importante resaltar que el aprendizaje centrado en el

estudiante [29] no disminuye la carga de trabajo del docente, sino que la direcciona en actividades tutoriales y de retroalimentación. Sobre lo expresado, una ventaja del modelo es que los grupos de trabajo dividen el curso en "n" equipos (*n=cantidad de estudiantes / cantidad de grupos*) que, en parte, disminuye la carga de trabajo docente. Por otro lado, por las características mencionadas de los micro TP, éstos no deberían ser complejos (retomando el concepto de carga cognitiva). Por ejemplo, en la rúbrica de clases UML, asociamos al criterio de evaluación *CE-3 Uso correcto de la semántica*, el *micro TP 3* con el siguiente enunciado *"Utilizando una herramienta CASE, dibuje un diagrama de clases proponiendo un ejemplo de composición, agregación y generalización respetando los criterios establecidos. Proponga un enunciado cuya solución esté representada por el modelo de clases propuesto".* Además, como la heteroevaluación estará precedido por la auto y coevaluación, la calidad del TP integrador no requerirá de mucho esfuerzo ya que, las evaluaciones previas, en parte, garantizan un nivel de calidad razonable.

El proceso es asistido por la opción *Desarrollo de competencias conducido por rúbricas* del repositorio [26], que facilita la comunicación entre el docente y los estudiantes en el intercambio de archivos y en la retroalimentación docente.

Si bien todavía no se implementó el modelo, experiencias piloto dan cuenta de una respuesta adecuada por parte de estudiantes y docentes que, de todos modos, serán evaluadas en próximos trabajos donde se ajustará el proceso en virtud de los comentarios de docentes y estudiantes.

Por último, en las actividades de aprendizaje es importante determinar los tiempos asignados a cada estrategia para dimensionar el esfuerzo requerido al estudiante. Es necesario contemplar, no solo las horas presenciales del estudiante, sino también aquellas actividades autogestivas que, en suma, determinarán el esfuerzo total, en horas, en términos de créditos académicos. En el ejemplo, la carga de trabajo total, en actividades sincrónicas y autogestivas se corresponden, aproximadamente, a 12 horas de trabajo del estudiante.

4. Conclusión y trabajos futuros

El trabajo se fundamenta en el alineamiento constructivo, que subraya la coherencia entre los resultados de aprendizaje, el proceso de enseñanza y la evaluación, además, se basa en la teoría de la carga cognitiva, que considera la carga mental impuesta a los estudiantes. Utiliza conceptos del ABP como estrategia de aprendizaje y descriptores de la rúbrica, asociada al problema, como criterio de evaluación. En resumen, la propuesta vincula cada criterio de evaluación de la rúbrica con micros TP enfocados en aspectos específicos de las competencias que permitan que los estudiantes se concentren en ellos antes de integrarlos en un trabajo práctico que abarque la complejidad requerida por el resultado de aprendizaje establecido.

Se deberá explorar la posibilidad de ampliar el modelo para abarcar diferentes áreas de conocimiento, no exclusivamente temas informáticos, esto permitirá desarrollar y evaluar competencias en una variedad de contextos disciplinares. Si bien la propuesta presupone una mejora en el proceso de aprendizaje, es necesario realizar estudios empíricos para validar la efectividad del modelo y analizar si se produce una mejora

significativa en las habilidades y conocimientos de los estudiantes. También es menester realizar comparaciones entre esta propuesta y otros enfoques de desarrollo de competencias que permitan evaluar la eficacia relativa de diferentes estrategias y determinar cuál es la más adecuada para su desarrollo y evaluación. Por último, urgen establecer lineamientos para determinar el tiempo asignado a las actividades de aprendizaje para no sobrepasar la cantidad de horas establecidas en los créditos académicos asignados a la asignatura para que no impacten negativamente en la carga de trabajo del estudiante.

Referencias

1. Biggs, J.B.: Calidad del aprendizaje universitario. Educatio Siglo XXI. 22, 272 (2004).
2. Carlino, F.: De la alienación al alineamiento constructivo. Más allá de la trampa mecanicista. Cuaderno de Pedagogía Universitaria. 18, 58–70 (2021).
3. Oquendo-González, E.J., Velásquez-Pérez, Y., Rose-Parra, C., Cervera-Manjarrez, N.: El alineamiento constructivo para el desarrollo de la competencia científica. CIENCIAMATRIA. 8, 666–686 (2022).
4. Neil, C., Battaglia, N., De Vincenzi, M.: Marco metodológico para el diseño de rúbricas analíticas. Edutec Revista Electrónica. 80, 198–215 (2022).
5. Neil, C., Battaglia, N., De Vincenzi, M.: La matriz de competencias como herramienta para orientar la escritura de resultados de aprendizaje. In: Congreso de Tecnología en Educación y Educación en Tecnología. TE&ET 2023, Hurlingham (2023).
6. Acuña, M.G., Sosa, N.M.: Experimentando prácticas de enseñanza: El ABP, sus implicancias para el desempeño del rol del tutor. Revista de Ciencia y Tecnología. 63– 68 (2017).
7. Sánchez, S.N.D., Sotelo, J.P.A.: Perfil de egreso en los estudiantes de ingeniería: aportes significativos de la metodología del Aprendizaje Basado en Problemas. SCIÉNDO. 24, 35–43 (2021).
8. Espinoza Freire, E.E.: El aprendizaje basado en problemas, un reto a la enseñanza superior. Conrado. 17, 295–303 (2021).
9. Gil Galván, M. del R., Martín Espinosa, I., Gil Galván, F.J.: Percepciones de los estudiantes universitarios sobre las competencias adquiridas mediante el aprendizaje basado en problemas. Educación XX1: revista de la Facultad de Educación. (2021).
10. CAETI: Centro de Altos Estudios en Tecnología Informática, https://caeti.uai.edu.ar/, last accessed 2023/07/18.
11. Tobón, S.T., Prieto, J.H.P., Antonio, J., Fraile, G., Hall, P.: Secuencias didácticas: aprendizaje y evaluación de competencias. (2010).
12. Universidad Abierta Interamericana: Modelo curricular para el diseño y ajuste de planes de estudio de la UAI, https://uai.edu.ar/media/122314/lineamiento-devicerrector%C3%ADa-acad%C3%A9mica-n%C2%BA-119.pdf.
13. De Miguel Díaz, M.: Modalidades de enseñanza centradas en el desarrollo de competencias: Orientaciones para promover el cambio metodológico en el Espacio Europeo de Educación Superior. (2005).
14. Torres-Sanz, V., Tramullas, J., Sanguesa, J.A., Garrido, P., Martinez, F.J.: Rúbricas como estrategia de evaluación en entornos TICS [Rubrics as an evaluation strategy in ICT environments]. (2017).
15. Alsina Masmitjà, J.: Rúbricas para la evaluación de competencias. Barcelona: Octaedro. Recuperado el. 19, 2018 (2013).

16. Battaglia, N., Neil, C., De Vincenzi, M.: Software Engineering Competence-Based Learning in Collaborative Virtual Environments. V IEEE Congreso Mundial de Educación en Ingeniería (EDUNINE2021). (2021).
17. Valverde Berrocoso, J.: El uso de e-rúbricas para la evaluación de competencias en estudiantes universitarios: Estudio sobre fiabilidad del instrumento. REDU: Revista de Docencia Universitaria. (2014).
18. Sweller, J.: Cognitive load during problem solving: Effects on learning. Cogn Sci. 12, 257–285 (1988).
19. Van Merriënboer, J.J.G., Sweller, J.: Cognitive load theory in health professional education: design principles and strategies. Med Educ. 44, 85–93 (2010).
20. Kalyuga, S., Sweller, J.: Rapid dynamic assessment of expertise to improve the efficiency of adaptive e-learning. Educational Technology Research and Development. 53, 83–93 (2005).
21. Olivares, S.L., Heredia Escorza, Y.: Desarrollo del pensamiento crítico en ambientes de aprendizaje basado en problemas en estudiantes de educación superior. Revista mexicana de investigación educativa. 17, 759–778 (2012).
22. Bloom, B.S., Engelhart, M.D., Furst, E.J., Hill, W.H., Krathwohl, D.R.: Handbook I: cognitive domain. New York: David McKay. (1956).
23. Gutiérrez, J., la Puente, G., Martínez, A., Piña, E.: Aprendizaje basado en problemas: un camino para aprender a aprender. Colegio de Ciencias y Humanidades, Universidad Nacional Autónoma De México, UNAM. CDMX, México. Págs. 1–160 (2012).
24. Pimienta Prieto, J.H.: Secuencias didácticas: aprendizaje y evaluación de competencias en educación superior. Bordón: revista de pedagogía. (2011).
25. Aguado, M.E.G., Llorente, C.G.: Secuencias didácticas desde un enfoque de desarrollo de competencias. Ikastorratza, e-Revista de didáctica. 3 (2019).
26. UAI Rubrics, http://case.uai.edu.ar/rubrics.
27. Höffler, T.N., Leutner, D.: Instructional animation versus static pictures: A metaanalysis. Learn Instr. 17, 722–738 (2007).
28. OMG: Object Management Group: OMG, https://www.omg.org/.
29. Delgado Martínez, L.M.: Aprendizaje centrado en el estudiante, hacia un nuevo arquetipo docente. Enseñanza & Teaching: Revista Interuniversitaria de Didáctica. 139– 154 (2019).

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Desarrollo de Sistemas Inteligentes: experiencias en capacitación

Analía Amandi[1,2], Nelson Acosta[1], Marcelo Campo[1,2]

[1] Núcleo de Inteligencia Comportamental Empresarial – Universidad Nacional del Centro de la Pcia. de Buenos Aires (UNICEN)
Campus Universitario – Tandil – Buenos Aires – Argentina
[2] CONICET
{amandi,nacota,mcampo}@exa.unicen.edu.ar

Resumen. El desarrollo de sistemas inteligentes implica utilizar componentes que determinan su comportamiento en un contexto de incertidumbre y autonomía. Este paper describe experiencias de capacitación en desarrollo de sistemas inteligentes desde el punto de vista de la programación, del diseño y del proceso de desarrollo de software. Aplicando aprendizaje basado en proyectos, aprendices son capacitados en temáticas de inteligencia artificial, la cual aplican luego en proyectos de software más complejos afianzando los conceptos relativos al desarrollo de sistemas inteligentes. Los resultados han sido exitosos tanto en relación a la capacitación específica en inteligencia artificial como en su inserción en el proceso de desarrollo.

Keywords: Artificial Intelligence, Intelligent Agents, Chatbots, Design, Software Engineering

1. Introducción

Al mismo tiempo que el desarrollo de aplicaciones de inteligencia artificial por la industria de software está en continua expansión, van surgiendo cuestionamientos en relación a que si las técnicas en ingeniería de software actuales satisfacen las necesidades de los sistemas inteligentes [Shaw and Zhu, 2022]. Simultáneamente, se ha focalizado en la educación en inteligencia artificial como eje esencial para la capacitación de usuarios y desarrolladores [Su and Zhong, 2022].

Mientras se modifican los contenidos de escuelas primarias y secundarias integrando a los conceptos de programación, conceptos de inteligencia artificial [Touretzky et. al., 2019] [Su et. al., 2022], las universidades de nuestra región aún discuten sobre los contenidos de inteligencia artificial para los profesionales de informática, y usualmente los separan de la ingeniería de software. Particularmente se observa que, cuando se trata el tema de sistemas inteligentes, se enfoca casi absolutamente en aplicaciones centradas en *machine learning* a partir de grandes datos, dejando de lado el resto de las características del área.

Estos avances en la tecnología de software generan necesidades de capacitación de profesionales en la industria de software. Es importante enfocar en capacitaciones efectivas para resguardar la productividad de esta industria. Recordando, por ejemplo, los cambios producidos por la inserción abrupta de la programación orientada a objetos en la industria, se generaron propuestas para su rápida adaptación a estos cambios

[Amandi et. al. 1993]. Ahora es el tiempo de enfocarnos en la inserción de la inteligencia artificial al desarrollo de software.

El desarrollo de sistemas inteligentes presenta características distintivas. Éstas caracterizan la inteligencia del sistema o de algunos de sus componentes. La más conocida es la capacidad de aprender, lo cual puede ocurrir a partir de grandes datos o de un pequeño conjunto de datos, pero en ambos casos se tendrá que trabajar con conceptos como incertidumbre, grado de certeza, explicaciones de lo aprendido. La toma de decisiones, lo que puede ocurrir a partir del resultado del proceso de aprendizaje, en la cual la capacidad de explicar las decisiones tomadas es relevante. La toma de decisiones se asocia con la autonomía, ya que se puede tanto sugerir cursos de acción como ejecutarlos autónomamente. La interoperabilidad cuando estas entidades inteligentes hacia objetivos mutuamente beneficiosos. Las características mencionadas generan cambios en el desarrollo, particularmente dificulta el control de las respuestas esperadas.

En este contexto, en el cual se evidencia las urgencias en la capacitación de los profesionales de informática en el desarrollo de sistemas inteligentes, se propone en este artículo un enfoque de capacitación basado en la utilización de Aprendizaje Basado en Proyectos desde la introducción de los conceptos básicos de inteligencia artificial hasta su utilización en proyectos de desarrollo de software inteligente. Este enfoque ha sido aplicado durante cinco años, y en este artículo se realiza un análisis de sus resultados. Esta aplicación del enfoque de capacitación es materializada a partir de experiencias realizadas dentro de un contexto universitario en tres cursos fuertemente conectados, uno que introduce conceptos de inteligencia artificial a partir de la programación, uno que se enfoca en el diseño de sistemas de software y otro que se enfoca en el proceso de desarrollo de software.

El Aprendizaje Basado en Proyectos es una estrategia metodológica de enseñanza en el cual la experimentación práctica es el elemento central. De esta manera, se generan retos en un contexto cercano a la realidad del desarrollo de software, generando en los alumnos la necesidad de búsqueda de soluciones, integrando conocimientos adquiridos en diferentes materias, y detectando la necesidad de aprender otros para resolver los retos que enfrenta. Este aprendizaje activo permite incorporar los conceptos de inteligencia artificial en el desarrollo de software, analizando las particularidades relativas a la autonomía, la incertidumbre, y las necesidades particulares de aceptación del usuario relacionadas con la explicabilidad.

Este artículo se estructura de la siguiente forma. La sección 2 presenta la propuesta de capacitación en Inteligencia Artificial en el contexto de Ingeniería de Software. En la sección 3 se enfoca en la introducción de inteligencia artificial desde el punto de vista de ingeniería. En la sección 4 se introduce inteligencia artificial en el diseño de sistemas de software. En la sección 5 se considera la inteligencia artificial en el proceso de desarrollo de software. En la sección 6 se analizan las experiencias. Finalizando en la sección 7 con las conclusiones.

2. Inteligencia Artificial en Ingeniería de Software

La introducción al mundo de la inteligencia artificial sigue siendo llevada a cabo como un mundo diferente y separado de la ingeniería de software. Se analizan conceptos básicos como razonamiento lógico, y diferentes técnicas de machine learning. Luego, no son generalmente incluidos o relacionados con la ingeniería de software. Esta situación viene siendo arrastrada de otras épocas en las que la inteligencia artificial era temas de investigación no siendo relevantes para la industria de software. Pero esto ha cambiado, y en los últimos años la evolución de los cambios es vertiginosa.

En estos momentos de la historia, en el cual la industria de software necesita de profesionales capacitados en la Ingeniería de Inteligencia Artificial, en el cual se cuentan con bibliotecas con implementaciones de diferentes arquitecturas de redes neuronales, redes bayesianas, entre otras, no capacitar a los profesionales en el desarrollo de sistemas inteligentes con todos los conceptos que esto involucra.

El Aprendizaje Basado en Proyectos (ABP) en el enfoque adoptado y reportado en este artículo, el cual ha sido exitosamente aplicado en el área de software [Gary 2015], y en estas experiencias se aplica en el desarrollo de sistemas inteligentes desde la introducción misma de los conceptos básicos de inteligencia artificial.

Aplicando ABP, se puede trabajar con los aprendices en cómo utilizar una determina biblioteca, e ir aprendiendo cómo un componente puede aprender a partir de un conjunto de datos, si esos datos son suficientes, y particularmente, palpar que la inteligencia artificial no es magia.

La Figura 1 muestra un esquema del enfoque adoptado. La fecha muestra una secuencia de los conceptos de Programación, Diseño y Procesos, los cuales representan tres cursos sucesivos. La rueda alrededor del concepto de Inteligencia Artificial define las características que tienen que ser introducida en la secuencia Programación-Diseño-Procesos. Estas características son: aprendizaje, incertidumbre, autonomía y explicabilidad.

De esta manera, en curso de programación se introduce inteligencia artificial, pero desde la práctica, programando. Se utiliza el concepto de objeto de la programación orientada a objetos como base, convirtiéndolo en inteligente [Amandi y Price, 1998]. Aquí se utiliza el recurso de la programación exploratoria, la cual define cuando: (i) un programador escribe código para construir prototipos o experimentar con diferentes ideas, y (ii) el programador no sólo intenta coincidir con la especificación sino que el objetivo es abierto y evoluciona a través del proceso de programación [Kery y Myers, 2017].

En el curso de diseño, estas entidades inteligentes ya conocidas son parte de los retos del aprendizaje basado en proyectos, aprendiendo no sólo los conceptos de diseño sino que se analiza cuál es el impacto del diseño de componentes inteligentes en relación al aprendizaje, la incertidumbre, la autonomía y la explicabilidad. Particularmente, se analiza en las experiencias con el usuario la relevancia de las explicaciones de las sugerencias y principalmente de las acciones autónomas de los componentes inteligentes.

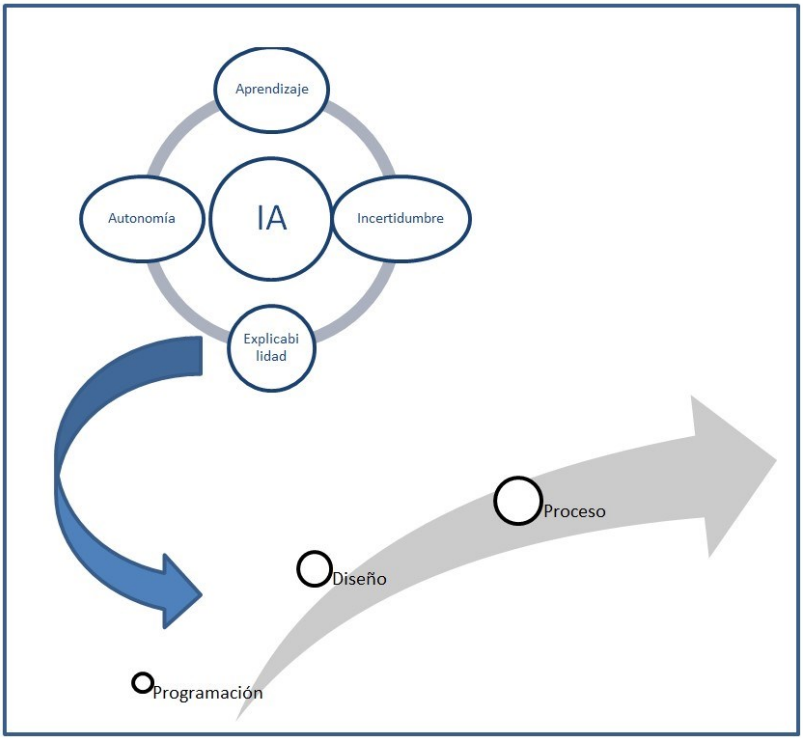


Figura 1. Esquema del enfoque de capacitación propuesto

En el curso donde se enfoca en el proceso, siempre en forma práctica, y en este caso utilizando un desarrollo completo de un sistema que incluye componentes inteligentes, en el cual todo el proceso es analizado. Aquí, todos los conceptos envueltos en Inteligencia Artificial son analizados en el contexto global del desarrollo mismo.

A continuación se desarrollará cada etapa Programación-Diseño-Proceso en las siguientes tres secciones.

3. Comprensión de Inteligencia Artificial

La primera etapa de la capacitación tiene por objetivo que los aprendices comprendan el concepto de inteligencia artificial. Esto implica no sólo la comprensión del concepto de inteligencia artificial en forma abstracta, sino de su materialización concreta en entidades inteligentes que pueden aprender, tomar decisiones, actuar, e interactuar con otras entidades inteligentes.

Para comprender que es un sistema inteligente, lo primero a lo que se enfrentan los aprendices es a un conjunto de sistemas inteligentes que están en funcionamiento, y que fueran construidos por la industria de software. Esto les permite ubicar la inteligencia artificial como de relevancia actual.

Una vez que los aprendices perciben qué es un sistema inteligente, se pasa a construir entidades inteligentes. Las entidades inteligentes son construidas por los aprendices a partir de chatbots. Los chatbots pueden imitar una conversación humana, pudiendo ser útiles en educación, recuperación de información, negocios, comercio electrónico, entre otros [Adamopoulou y Moussiades, 2020].

Los chatbots son definidos como la entidad base para la construcción de entidades a las que conoceremos como agentes inteligentes. Los agentes inteligentes [Wooldridge y Jennings, 1995] son entidades que exhibe características distintivas como autonomía. Para esta construcción, se utiliza un framework para desarrollo de chatbots. Durante la experiencia aquí reportada, con los tres grupos distintos de aprendices, se han utilizado diferentes frameworks, mayoritariamente RASA y IBM Watson Assistant. En la Figura 2, se muestra parte del código RASA y la visualización de la conversación a la que corresponde.

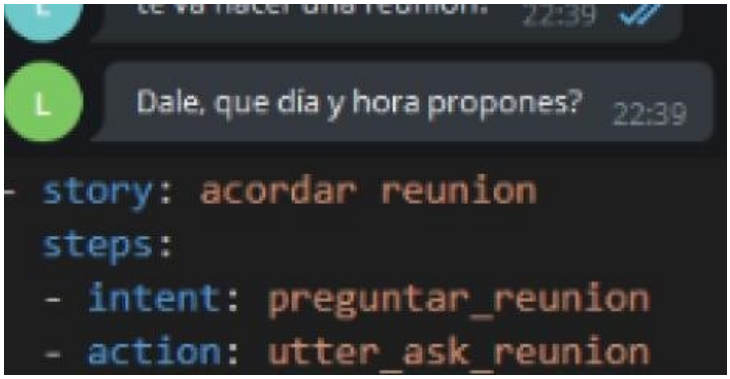


Figura 2. Chatbot

Dentro de ABP, el enfoque utilizado para analizar los conceptos de inteligencia artificial es el de la programación exploratoria. En consecuencia, se parte de una especificación, y se explora con mente abierta los alcances que la inteligencia artificial nos habilita. En las experiencias, se ha explorado en la construcción de agentes inteligentes que representen docentes, empleados, negocios, etc. En algunos de los casos con interacciones con varios humanos, como un coordinador de un grupo de trabajo [Amandi et.al., 2022].

Estos chatbots, vistos como agentes inteligentes, se los va construyendo explorado:

- Personalizar la interacción con el usuario: para esto tiene que recolectar datos del usuario, generar un perfil y adaptar el comportamiento dinámicamente en función del mismo.
- Adaptar la interacción por aspectos emocionales detectados: aquí se incorporan conceptos de computación afectiva a partir de texto. Se reconocen emociones en las interacciones y se responde en consecuencia.

Por otro lado, la propia construcción del chatbot permite a los aprendices comprender el concepto de machine learning. Por ejemplo, el framework RASA utiliza una red neuronal recurrente para la detección de intenciones de comunicación en los

chatbots. Los aprendices ingresan ejemplos para definir y así reconocer cada intención de comunicación. Así, pueden experimentar cuando el entrenamiento es suficiente o no, para el reconocimiento de una intención, y cuando se confunde con intenciones con algún grado de similaridad. Consecuentemente, se alcanzan cuatro objetivos simultáneamente: (i) comprender el concepto de *machine learning*, (ii) comprender la importancia de los datos de entrenamiento, (iii) comprender cómo funcionan los datos de prueba y las necesidades de reentrenamiento, y (iv) cómo utilizar una implementación de una red neuronal sin ser especialistas, más aún, sólo teniendo un conocimiento arquitectónico del tema.

4. Diseño de software con componentes inteligentes

Finalizada la etapa de introducción de inteligencia artificial a partir de la programación exploratoria, los aprendices son introducidos en diseño de sistemas de software, pero considerando que estos sistemas pueden incluir entidades inteligentes.

A los aprendices, siguiendo al ABP, se les plantea los requerimientos de un sistema que incluye entidades inteligentes. Así, agrupados ahora en equipos de trabajo, tendrán el reto del diseño del sistema objetivo. En las experiencias reportadas, se les plantea el reto de construir un ambiente de realidad virtual con una finalidad específica, en el cual avatares inteligentes y autónomos serán parte del mismo.

Los ambientes de realidad virtual [Scott et.al., 2017] presentan un reto para los aprendices por su complejidad, teniendo que trabajar con diferentes frameworks, integrándolo. Adicionalmente, agregando más componentes de la realidad, cada una de las experiencias, se iniciaron desde el código desarrollado por el grupo anterior.

Los chatbots, ahora ya conocidos por los aprendices, convertidos en agentes inteligentes, se incorporan en un ambiente de realidad virtual. En los años de las experiencias, se han ido construyendo diferentes sistemas en este entorno. En la Figura 3, se muestra dos de los *bots*, que son aquellos chatbots pero visualizados a través de una imagen humana.



Figura 3. Incorporando agentes inteligentes en un ambiente de realidad virtual

Los aprendices se enfocan en el diseño, generando prototipos para su validación. Los aprendices, reunidos en equipos, deben esforzarse en la funcionalidad con calidad, diseñando con miras a cumplir con requisitos arquitectónicos significativos como usabilidad, confiabilidad, seguridad, rendimiento y escalabilidad.

Los objetivos en esta etapa son, técnicamente hablando, la comprensión práctica de estilos arquitectónicos, atributos de calidad, patrones de diseño, frameworks, etc. Esto se alcanza aplicando ABP, mediante el desarrollo de un sistema que presenta retos desafiantes. A estos retos, se le suma la incorporación de agentes inteligentes, que utilizan algoritmos de *machine learning*. Sin embargo, el objetivo primario sigue siendo el diseño de sistemas de software.

Un ejemplo de estos desarrollos se reporta en [Campo et.al., 2021b], en el cual el ambiente de realidad virtual es un ambiente virtual de desarrollo de software. Esta experiencia muestra su complejidad en la utilización de diferentes frameworks como Unity para los componentes de realidad virtual, y RASA para los aspectos conversacionales de los agentes inteligentes.

Otro ejemplo de sistema desarrollado en esta instancia es un ambiente de realidad virtual para capacitación, el cual es conectado con cursos Moodle. A su vez, se ha experimentado con el framework Moodle para incorporar agentes que permitan su personalización [Campo et.al., 2021a].

Al poner el foco de atención en los conceptos de diseño de sistemas de software, la incorporación de agentes inteligentes se analiza en este contexto, siendo natural la integración de inteligencia artificial al desarrollo de sistemas. A partir de la propia práctica, surgen interrogantes relativos a los datos variables asociados a los agentes inteligentes, y particularmente a la aceptabilidad por parte de los usuarios potenciales del sistema objetivo.

En los ejemplos mencionados, se analizó la experiencia del usuario. En el caso del ambiente virtual de desarrollo de software, los propios aprendices analizaron los resultados. En el caso del ambiente virtual de capacitación, otro de los grupos de aprendices tomó el rol de usuario, con la finalidad de evaluar la experiencia del usuario y proveer *feedback* a los desarrolladores. En este análisis, es importante resaltar el análisis de la aceptabilidad de los usuarios de los agentes inteligentes.

5. Proceso de desarrollo de software

En este punto, los aprendices ya tienen conocimiento de programación, de inteligencia artificial y de diseño de sistemas de software. El objetivo ahora es preparar a los aprendices en el proceso mismo de desarrollo de software. Aquí, se continúa trabajando en equipo, y el trabajo en equipo pasa a tener mayor relevancia en esta capacitación.

Las experiencias realizadas con los grupos están orientadas en forma práctica del conocimiento de procesos de desarrollo de software. SAFe (*Scaled Agile Framework*) [Kane, 2018] fue la guía de estas experiencias. En el *framework* SAFe, la agilidad no debe ser tan importante que se pague con calidad. Hay cinco dimensiones clave de calidad: calidad de flujo, arquitectura y diseño, calidad de código, calidad del sistema y calidad del *release*.

Los equipos de trabajo son realmente relevantes en el contexto de SAFe. Éste pone especial atención al liderazgo. Los líderes son los responsables de crear el contexto necesario para adoptar todos los valores definidos como importantes.

En la práctica, los equipos tendrán la responsabilidad del desarrollo de un sistema inteligente. Los equipos estarán conformados por desarrolladores con diferente grado de capacitación, con la finalidad de generar un entorno lo más real posible.

En este contexto, la inteligencia artificial es inserta en el proceso de desarrollo en sí. Los equipos, en su capacitación misma, participan en cada paso de desarrollo del sistema siguiendo SAFe, quedando registrado de esta manera cómo influencia el desarrollo de entidades inteligentes cada etapa. Estos registros permiten analizar la experiencia por cada equipo al final la misma.

En este contexto, se da continuidad con los proyectos desarrollados (a nivel de prototipo) en la segunda etapa. Por lo que se continúa con el desarrollo de ambientes de realidad virtual. Particularmente, se destaca la evolución del ambiente de realidad virtual para desarrollo de software [Scott y Campo, 2021], el cual permite ver y experimentar en el mismo ambiente el desarrollo.

6. Análisis de las experiencias

Se reporta en artículo, cinco años de experiencias con grupos de aprendices. Cada grupo de 25 aprendices cada uno fue observado en cada paso, y se fueron registrando un conjunto de parámetros.

Cada experiencia completa se desarrolla en tres años consecutivos, y está compuesta de tres pasos. Cada paso se ejecuta en cuatro meses. Los aprendices inician la capacitación con edades entre 1921 años, y el grupo está compuesto por un promedio de 93% de varones y un 7% de mujeres. En los cinco años de experiencias: tres grupos cumplieron todo el ciclo, y son los que se reportan en este artículo. Los pasos son:

6.1 Primer paso: Inteligencia artificial desde la Programación Exploratoria

En este primer paso, se registra: (i) desarrollo de *chatbots*; (ii) desarrollo de agentes inteligentes (comprensión básica, grado de innovación, manipulación de emociones, personalización); (iii) *machine learning* (comprensión básica, entrenamiento, re-entrenamiento, ambigüedades).

En la Tabla 1, se presentan estas variables para los tres grupos de estudio.

La primera variable de la Tabla 1 se refiere al desarrollo mismo de los chatbots, y refiere a la calidad de los *chatbots* desarrollados. La calidad de cada chatbot de cada uno de los 25 miembros del grupo fue calificado con un valor de 1 a 10, siendo 1 la menor calificación y 10 la máxima. La calidad involucra el manejo del diálogo, incluyendo el manejo de contextos en el propio diálogo. Los valores promedio mostraron una excelente comprensión en todos los grupos.

Los *chatbots* se extendieron para convertirse en Agentes Inteligentes. Se evalúa en esta sección la comprensión de qué es un agente inteligente, qué tan innovativo fue el enfoque de cada aprendiz, el manejo de emociones, y el grado de personalización.

También fue calificado entre 1 y 10, siendo 10 la mejor calificación. Los valores promedio que vemos en la Tabla 1, muestra que la comprensión general del concepto de agente inteligente es buena, y que se necesita trabajar más la personalización y el manejo de emociones. Más adelante, en la Tabla 3, finalizadas las tres etapas, se muestra la mejora en el desarrollo de agentes inteligentes.

Tabla 1. Variables consideradas en la evaluación del paso Inteligencia Artificial desde la Programación Exploratoria

		Grupo 1	Grupo 2	Grupo 3
Desarrollo de chatbots		8.38	9.38	9.48
Agentes Inteligentes	**Comprensión básica**	8.08	8.76	7.52
	Innovación	6.52	6.66	6.26
	Emociones	4.6	4.4	4.48
	Personalización	5.6	5.4	5.48
Machine Learning	**Comprensión básica**	7.3	7.58	7.34
	Entrenamiento	7.5	7.7	7.48
	Re-entrenamiento	6.94	6.5	6.48
	Ambigüedades	6.14	6.16	5.98

En el desarrollo de los *chatbots*, se definen intenciones de comunicación, ingresando datos de entrenamiento para cada una de ellas. A partir de aquí, se analiza con los aprendices la problemática de *machine learning*, mediante el análisis de cuatro variables: comprensión básica, entrenamiento, reentrenamiento, y manejo de ambigüedades. Lo que podemos analizar de los resultados de la Tabla 1, es que si bien los aprendices construyeron chatbots con facilidad, los conceptos subyacentes relativos a machine learning necesitan ser trabajados. Lo cual realizarán en las siguientes etapas, para afianzar estos conocimientos en el contexto del desarrollo de software.

6.2 Segundo paso: Diseño de Sistemas Inteligentes

En este segundo paso, se registra: (i) manejo de experiencia del usuario (aceptabilidad, confiabilidad); (ii) arquitectónico; (iii) manejo de evolución del aprendizaje.

En la Tabla 2, se presentan estas variables para los tres grupos de estudio.

Tabla 2. Variables consideradas en la evaluación del paso Diseño de Sistemas Inteligentes

		Grupo 1	Grupo 2	Grupo 3
Experiencia del usuario	**Aceptabilidad**	7.46	7.58	7.44
	Confiabilidad	7.64	8.06	8.44
Arquitectónico		8.58	8.36	8.02
Evolución del aprendizaje		8	8.6	7.98

Lo primero que vemos en esta Tabla 2, es la evolución desde la Tabla 1. En el tratamiento arquitectónico de las entidades inteligentes vemos la evolución del

desarrollo de agentes inteligentes. Del análisis de la evolución del aprendizaje, vemos la maduración en la temática de *machine learning*.

En relación a las dos primeras variables de la Tabla 2, que evalúa el diseño de entidades inteligentes en el contexto de las experiencias de los usuarios, se analiza la aceptabilidad y la confiabilidad. A pesar que los resultados muestran valores altos, variando de 7.5 a 8, se ha notado las dificultades de los aprendices en dejar de lado su propia percepción para sólo tomar en cuenta las percepciones y opiniones de los usuarios.

6.3 Tercer paso: Proceso de Desarrollo de Sistemas Inteligentes

En este tercer paso, se registra: (i) desarrollo de agente inteligentes; (ii) cálculos de tareas relativas a IA; (iii) satisfacción del equipo.

En la Tabla 3, se pueden observar estas variables en los tres grupos de estudio. En esta Tabla, vemos la evolución final de la capacitación en la primera variable, la evolución es positiva y significativa.

La segunda variable analizada es la correspondiente a los cálculos de tiempos para las tareas relativas a inteligencia artificial. Los números claramente nos informan de las dificultades en calcular los tiempos de desarrollo de componentes que involucren conceptos de inteligencia artificial. Esto se vio reflejado en las reuniones diarias de avance. Se destaca esta variable en el desarrollo de sistemas inteligentes ya que representa la causa principal de la siguiente variable, la que mide la satisfacción del equipo de trabajo.

Los valores de satisfacción del equipo de trabajo rondan en 7 puntos. En las reuniones de evaluación de los avances de cada equipo, el entrenamiento como parte del proceso de *machine learning*, particularmente en el armado de perfiles, generó incertidumbre. Estos resultados son interesantes como punto de partida para un mayor análisis de la integración de inteligencia artificial e ingeniería de software.

Tabla 3. Variables consideradas en la evaluación del paso
Proceso de Desarrollo de Sistemas Inteligentes

	Grupo 1	**Grupo 2**	**Grupo 3**
Desarrollo de agentes inteligentes	9.08	9.62	8.52
Cálculo de tareas relativas a IA	5.6	5.86	6.2
Satisfacción del equipo	7.08	6.92	7.2

7. Conclusiones

Estamos cursando un momento de cambios en las posibilidades tecnológicas de la industria de software. Las mayores potencias tomaron la delantera hace cinco años atrás en términos de educación desde la escuela primaria, capacitando no sólo a los futuros profesionales sino también a los futuros usuarios de innovaciones en tecnología. Los avances vertiginosos de la inteligencia artificial nos llevaron a este punto.

En este contexto, se presenta aquí una experiencia conjunta entre la inteligencia artificial y la ingeniería de software en términos de capacitación de profesionales. Estas experiencias están encuadradas en el Aprendizaje Basado en Proyectos, el cual es entrelazado con la Programación Exploratoria como introducción a la inteligencia artificial. La programación exploratoria es clave en situaciones donde los atributos claves de exploración son la flexibilidad, el descubrimiento y la innovación [March, 1991]. Hoy necesitamos preparar a los profesionales como personas que posean la flexibilidad suficiente para explorar y así generar sistemas innovativos con las nuevas ideas, técnicas y frameworks generados a partir de la inteligencia artificial, pero dentro de procesos claros de desarrollo de software. Creemos que reproduciendo estas prácticas, posibilitará acelerar la inserción de la industria de sistemas inteligentes en el mercado internacional.

Referencias

1. Adamopoulou, E., Moussiades, L. (2020) An overview of chatbot technology. Artificial Intelligence Applications and Innovations, (584), p.373-383.
2. Amandi, A., Leonardi, M.C., Lopez, M.J., Prieto, M. and Rossi, G. (1993) Learning object-oriented concepts with multimedia technology. OOPSLA Addendum 1993: 13-16.
3. Amandi, A. and Price, A. (1998) Building Object-Agents from a Software Meta-Architecture. In Advances in Artificial Intelligence, 14th Brazilian Symposium on Artificial Intelligence, SBIA´98, Porto Alegre, Proceedings. Lecture Notes in Computer Science 1515, p. 21-30, Springer.
4. Amandi, A., Campo, M., Faiella, S. and Vibart, E. (2022) Coordinación de Grupos mediante Bots Conversacionales. Simposio Argentino de Inteligencia Artificial ASAI, p.88-91.
5. Amershi, S., Begel, A., Bird, C., DeLine, R., Gall, H., Kamar, E., Nagappan, N., Nushi, B. and Zimmermann, T. (2019) Software engineering for machine learning: a case study. IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice.
6. Campo, M., Amandi, A. and Biset, J. (2021a) A software architecture perspective about Moodle flexibility for supporting empirical research of teaching theories. Education and Information Technologies, 26(1), p.817-842.
7. Campo, M., Amandi, A., Acosta, N. and Guerrero, M. (2021b) AgileTalk: A Conversational Virtual Reality Synthetic Platform for Training in Agile Processes. Argentine Symposium on Software Engineering, ASSE.
8. Chenoweth, S., Linos, P. (2022) Crafting an Udergraduate Course at the Intersection of Machine Learning and Software Engineering. IEEE Frontiers in Education Conference.
9. Gary, K. (2015) Project-based learning. Computer. 48(9) IEEE, p.98-100.
10. Hulten, G. (2018) Building intelligent systems: a guide to machine learning engineering. Springer.
11. Kane, L. (2018) SAFe PI Planning: A step-by step guide. 136 pag.
12. Kery, M.B. and Myers, B. (2017) Exploring Exploratory Programming. 2017 IEEE Symposium on Visual Languages and Human-Centric Computing.
13. March, J.G. (1991) Exploration and exploitation in organizational learning. Organ. Sci., 2(1), p.71-87.
14. Pantic, M. Zwitserloot, R. and Grootjans, R.J. (2005) Teaching introductory artificial intelligence using a simple agent framework. IEEE Transactions on Education 48(3)
15. Scott, E., Soria, A. and Campo, M. (2017) Adaptive 3D Virtual Learning Environments – A Review of the Literature. IEEE Trans. Learn. Technol. 10(3), p.262-276.

16. Scott, E. and Campo, M. (2021) An adaptive 3D virtual learning environment for training software developers in scrum. Interactive Learning Environment. En prensa. DOI: 10.1080/10494820.2021.1999985.
17. Shaw, M. and Zhu, L. (2022) Can software engineering harness the benefits of advanced AI?. IEEE Software 36(6), p.99-104.
18. Su, J. and Zhong, Y. (2022) Artificial Intelligence (AI) in early childhood education: Curriculum design and future directions. Computers and education: Artificial Intelligence, vol.3.
19. Su, J., Zhong, Y., Ng, D. A meta-review of literature on educational approaches for teaching AI at K-12 levels in the Asia-Pacific region. Computers and education: Artificial Intelligence, vol.3.
20. Touretzky, D. Garden-McCune, C., Martin, F. and Seehorn, D. (2019) Envisioning AI for K-12: What should every child know about AI?. AAAI 2019, p. 9795-9799.
21. Wooldridge, M., Jennings, N. (1995) Intelligent agents: Theory and practice. The Knowledge Engineering Review, 10(2), p.115-152.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Hacia una Educación Integral: Modelo Visual para la Interconexión de Conceptos

Lucas Videla[1] [0009-0009-3433-757X], Verónica Inés Aubin[1] [0000-0002-4981-6460], and Renata Silvia Guatelli[1] [0000-0001-6118-0162]

Departamento de Ingeniería e Investigaciones Tecnológicas
Universidad Nacional de La Matanza
Florencio Varela 1903, San Justo, Argentina
{lvidela,vaubin,rguatelli}@unlam.edu.ar

Resumen. La interdisciplinariedad se posiciona como una estrategia relevante en un mundo complejo y globalizado. En el ámbito educativo, se busca una formación integral del ingeniero informático que se adapte a requerimientos interdisciplinarios. El aprendizaje significativo y las neurociencias respaldan la importancia de conexiones sinápticas y esquemas cognitivos para un aprendizaje profundo y duradero. Sin embargo, la implementación de la interdisciplinariedad en el currículo enfrenta desafíos como la atemporalidad y la coordinación docente. Este trabajo presenta un modelo innovador que simula la complejidad del cerebro, permitiendo una articulación interdisciplinaria dinámica. Inspirado en el lenguaje de patrones, Zettelkasten y la brújula de ideas, el modelo ofrece una comprensión contextualizada de los conceptos, visualizados como un meta-grafo que crece en forma orgánica de acuerdo a las necesidades de los docentes. Este modelo facilita la creación de objetos de aprendizaje, brindando así una experiencia educativa más integral y significativa.

Keywords: interdisciplinariedad · articulación de contenidos · visualización de contenidos · aprendizaje significativo · calidad educativa

1. Introducción

En un mundo cada vez más complejo y globalizado, los desafíos y problemas que enfrentamos requieren de un enfoque integral y colaborativo que trascienda los límites tradicionales de las disciplinas individuales. La interdisciplinariedad busca establecer puentes entre diferentes campos del conocimiento, fomentando la integración de perspectivas, enfoques y metodologías diversas. Al promover la interacción entre disciplinas, se generan nuevas sinergias que permiten abordar de manera más eficaz los problemas complejos y promover avances significativos en la comprensión y solución de los desafíos científicos, tecnológicos, sociales y ambientales que enfrentamos en la actualidad. En el contexto de la educación en el nivel superior, la formación del ingeniero informático requiere de un perfil que se adapte a estos requerimientos interdisciplinarios. En la organización del currículo por disciplinas podemos distinguir articulaciones entre asignaturas de forma horizontal y vertical. La primera se refiere a aquellas asignaturas o módulos que deben ser cursados en el mismo ciclo. La articulación vertical se corresponde con el orden en que las asignaturas deben ser cursadas en diferentes ciclos [7]. Bajo el enfoque tradicional, cada asignatura se dicta como un compartimento estanco habitualmente sin que rebasen los límites de la propia

disciplina. Este enfoque educativo brinda al estudiante una mirada parcelada de una realidad que es mucho más compleja e interesante de lo que se le presenta. En la actualidad el sistema universitario adopta un enfoque basado en competencias y centrado en el estudiante [11], con la idea de lograr que sus estudiantes alcancen aprendizajes profundos y estables en el tiempo. La formación con base en competencias conlleva integrar disciplinas, conocimientos, habilidades, prácticas y valores [3].

El aprendizaje significativo es un objetivo clave en la educación. Al conectar nuevos conocimientos con la estructura cognitiva existente, fomentar la colaboración, trabajar con situaciones reales y permitir la construcción activa del conocimiento, se puede facilitar un aprendizaje más profundo, relevante y duradero para los estudiantes. Además, las neurociencias respaldan este enfoque al destacar la importancia de las conexiones sin ‘apticas y los esquemas cognitivos en el proceso de aprendizaje. El aprendizaje significativo consiste en transferir información a la memoria a largo plazo. Cuando se adquiere nueva información, el cerebro la almacena en forma de esquemas, que son estructuras cognitivas que constituyen la base del conocimiento [15]. Estos esquemas permiten organizar y recuperar la información para procesarla y relacionarla con nuevos conocimientos. En este sentido, al establecer múltiples conexiones entre conceptos relacionados, la capacidad de comprensión profunda se ve enriquecida, lo que permite aplicar y transferir el conocimiento a diferentes situaciones de la vida real.

La interdisciplinariedad en el ´ámbito educativo al integrar diferentes disciplinas y enfoques, logra establecer conexiones entre los conocimientos aislados, fomentando así una comprensión más profunda y contextualizada de los temas [9,12]. La incorporación de la interdisciplinariedad en las actividades curriculares plantea desafíos importantes, ya que requiere una construcción deliberada dado que no ocurre espontáneamente. La coordinación entre docentes de diferentes ´áreas puede ser complicada, especialmente debido a la falta de coincidencia en horarios y agendas. Además, la diferencia de vocabulario y enfoque entre materias puede dificultar la comprensión conjunta. Superar estas barreras es fundamental para lograr una integración efectiva de los conocimientos y promover el aprendizaje significativo y enriquecedor para los estudiantes.

Atendiendo a estos problemas y teniendo en cuenta que los conocimientos sin vinculación entre sí rompen la asimilación consciente de los conocimientos y habilidades [14], en este trabajo se presenta una novedosa forma para articular contenidos. A diferencia de la articulación tradicional horizontal y vertical, donde los contenidos se organizan de manera lineal y secuencial, la articulación propuesta simula la complejidad y flexibilidad del cerebro, permitiendo una integración interdisciplinaria más dinámica y profunda. Inspirados en el lenguaje de patrones, la metodología de Zettelkasten y la brújula de ideas, adaptándolas y resignificándolas en el contexto de la educación. Haciendo uso de las NTICS, se creó un modelo de visualización que permite una comprensión más profunda y contextualizada de los conceptos, visualizados como una red representada en un meta-grafo. El modelo presentado, aun trabajando desde una perspectiva curricular, facilita al docente la articulación de contenidos y la comprensión interdisciplinaria de los conceptos. Además, posibilita el establecimiento de nuevas conexiones para la creación de objetos de aprendizaje que aborden diferentes contextos. Al fomentar una comprensión más profunda y conexa, se propicia un aprendizaje más significativo y transferible, fortaleciendo así la formación integral de los estudiantes.

2. Marco Teórico

Los componentes constitutivos de la metodología propuesta en este trabajo son el Zettelkasten, la brújula de ideas, el lenguaje de patrones y la visualización espacial.

2.1 Zettelkasten

El término "Zettelkasten" se refiere a una técnica conocida como "caja de fichas" o "fichero" en alemán, desarrollada por Niklas Luhmann. En esta metodología, se toman notas breves y en palabras propias sobre lecturas o temas de interés. La singularidad y el potencial de esta técnica radican en la capacidad de establecer conexiones entre las fichas, facilitada por la naturaleza atómica de las ideas expresadas en ellas.

Luhmann observó que el valor de una idea depende en gran medida de su contexto, y sus contextualizaciones. Por lo tanto, surgió la inquietud de cómo una idea podrá relacionarse y contribuir a diversos contextos [1].

La construcción del repositorio de ideas se lleva a cabo de manera incremental, estableciendo vínculos entre cada idea y otras previamente identificadas, lo que permite descubrir conexiones significativas. Esta metodología fomenta la repetición espaciada para la construcción de conocimiento, ya que se revisan las ideas anteriores cada vez que se introduce un nuevo concepto que aporta un contexto novedoso. La repetición espaciada ha demostrado ser eficaz para el aprendizaje significativo [4,13].

2.2 Brújula de Ideas

Se ha popularizado una técnica denominada "brújula de ideas" con el objetivo de facilitar el descubrimiento deliberado de conexiones entre conceptos [16]. La base de esta técnica radica en la concepción de una *brújula conceptual*, que posibilita la identificación de relaciones significativas entre ideas, valiéndose de la analogía de los puntos cardinales.

Tseng propone un enfoque de vinculación con otras concepciones de la siguiente manera: hacia el norte, estableciendo conexiones con los conceptos desde los cuales se originó; hacia el sur, relacionándola con los conceptos a los que conduce; hacia el oeste, conectándose con ideas similares y afines; y finalmente, hacia el este, identificando los conceptos opuestos o competitivos que se contraponen a la idea objeto de análisis [18].

2.3 Lenguaje de Patrones

La concepción de patrones tiene raíces ancestrales, pero su formalización como un conjunto sistematizado de alternativas coexistentes fue desarrollada por Alexander. En su obra sobre diseño urbano, Alexander sostiene que la construcción de cualquier elemento no debe aislarse, sino que debe considerar y armonizar con su entorno para lograr coherencia y completitud [2].

En el ´ámbito del software, estas ideas fueron adoptadas para la creación de diversas publicaciones que clasifican y especifican patrones. La primera de ellas estableció una estructura reutilizable para la especificación de patrones, definiendo secciones que

caracterizan cada patrón y facilitan su comprensión, aplicabilidad, contexto y consecuencias de uso [10].

Por su parte, Coplien realizó un estudio sobre el lenguaje de patrones [6] y su relevancia donde destaca dos razones principales para la importancia del estudio y aplicación de patrones. En primer lugar, los patrones capturan y visibilizan prácticas recurrentes en un contexto que de otra manera serían indefinidas y difusas. En segundo lugar, los patrones revelan una estructura subyacente que, una vez identificada, permite establecer conexiones significativas entre conceptos que previamente parecían independientes y aislados. Este enfoque en patrones facilita la comprensión y mejora de soluciones en el campo del diseño y la implementación de software y, potencialmente, en otros campos de conocimiento.

2.4 Visualización Espacial

La visualización de información va más allá de la simple creación de gráficos o imágenes. Su objetivo es desencadenar un proceso cognitivo que potencie nuestras habilidades de comprensión al obtener conocimientos a partir de representaciones visuales. Mediante la visualización de información, es posible transformar datos abstractos, en representaciones visuales significativas.

Uno de los campos que abarca la Visualización de Información es la Visualización de Grafos, que direcciona el problema de visibilizar información estructural o relacional construyendo representaciones visuales geométricas de grafos o redes que son los modelos subyacentes en una gran cantidad de datos abstractos.

La representación visual de los grafos suele ser una práctica común, aunque no es suficiente simplemente conectar nodos entre sí para capturar toda la riqueza semántica de la información. Es fundamental enriquecer la representación visual con formas, estereotipos, ubicaciones espaciales, colores, tamaños y otros componentes para transmitir de manera efectiva la información relevante [17].

Además, las relaciones entre los elementos del grafo pueden tener atributos de información asociados, especialmente en conjuntos de datos multivariados. Los nodos y aristas del grafo pueden albergar propiedades adicionales en función de representaciones gráficas que refuercen la comprensión del conocimiento que estos quieren manifestar [5].

3. Desarrollo

En esta sección se detalla cómo fue el proceso de construcción del modelo, utilizando el marco teórico anteriormente descripto.

3.1 El lenguaje de patrones

El proceso de creación de este modelo inició con una descripción estandarizada de los conceptos que conforman el nodo de articulación. En concordancia con los principios propuestos por Coplien, se adoptó una estructura que facilitó tanto la

vinculación como la navegación entre los distintos nodos. Esta metodología proporcionó un marco conceptual sólido, el cual se vio enriquecido por diversos enfoques que ampliaron el proceso de análisis y comprensión del modelo.

A partir de la primera aproximación, se obtuvo un nodo estructurado que contó con secciones claramente definidas, lo que permitió detallar el concepto atómico bajo estudio. Además, se identificaron secciones correspondientes a enfoques específicos otorgados al mencionado concepto en el ´ámbito de determinadas asignaturas o contextos.

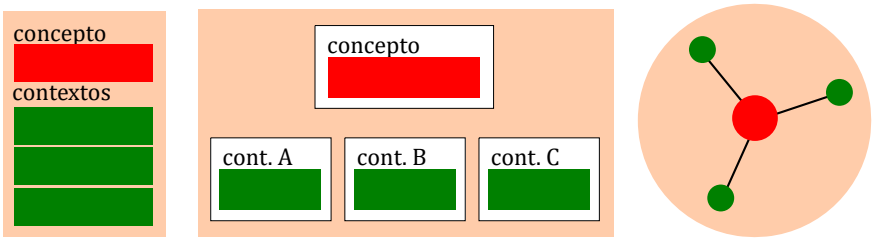
La representación física de este nodo de articulación adoptó la forma de una ficha, incorporando sucesivas secciones destinadas a señalar los distintos enfoques asignados al concepto, como puede verse en la Figura 1(a). Esta ficha resultó demasiado extensa, abarcando contenidos que pueden no ser de interés para todos los docentes.

3.2 Las ideas atómicas y el Zettelkasten

Desde la perspectiva del Zettelkasten se sugiere la implementación de un modelo flexible y reutilizable mediante la división de la estructura de la ficha en fichas específicas de segundo orden, junto con los enlaces correspondientes. Esta propuesta permitió separar el nodo en dos componentes distintos: en una parte se especificó el concepto concreto que representa, mientras que en fichas separadas se abordaron diversas aproximaciones al mismo desde distintos contextos, enriqueciendo así su contenido mediante resignificación. Esto puede verse en la Figura 1(b).

Esta subdivisión facilitó la separación de ideas atómicas, tal como se expone en el trabajo de Ahrens, las cuales adquieren significado al aplicarse a nuevos contextos. Además, esto permitió establecer relaciones interdisciplinarias entre las ideas, resultando esenciales para abordar la complejidad inherente al conocimiento.

Los contextos de aproximación abarcaron desde asignaturas completas, como es el caso de “la recursividad desde la perspectiva de la programación”, hasta áreas temáticas específicas dentro de una misma asignatura, como “la recursividad desde la programación funcional” o “la recursividad desde la programación estructurada”. La definición flexible de los contextos permitió adaptarlos según las necesidades particulares de los docentes, ajustando su granularidad para una ´optima comprensión y aplicación, como puede verse en la Figura 1(c).



(a) Ficha (b) Ficha desagregada (c) Nodo

Figura1: Ficha desagregada en sus componentes, y su visualización en forma de grafo.

Los enfoques contextuales desempeñaron un papel fundamental en la aplicación del lenguaje de patrones a los nodos de articulación. Mediante estos enfoques, se permitió la consideración del contexto en el cual cada nodo se encontraba inserto, lo cual confirió significado y relevancia específica a su función. El abordaje contextual del nodo de articulación conllevó a una mayor comprensión de su papel en el sistema, identificando las interconexiones con otros elementos y su adaptabilidad a diversas situaciones.

3.3 Tendiendo puentes entre contextos

La atemporalidad se presenta como un entorno propicio para fortalecer el aprendizaje de conceptos en nuevos contextos. Esta característica permite que asignaturas correlativas aborden temáticas similares, profundizando en los conceptos adquiridos recientemente. A diferencia de la articulación vertical entre asignaturas, esta aproximación es un modelo emergente que se adaptó según la necesidad de abordar la temática de interés en cada contexto específico.

En este sentido, se propuso el desarrollo de objetos de aprendizaje que interconectan enfoques sobre un mismo concepto. Por ejemplo, una actividad vincula el enfoque matemático del concepto de factorial con la perspectiva de programación sobre el mismo tema. Esquemáticamente, esto puede visualizarse en la Figura 2. De esta manera, se reafirma el concepto y se logra una repetición espaciada del primer contexto al integrar los conocimientos y competencias proporcionados por el nuevo enfoque.

Este modelo ofrece una valiosa oportunidad para que los docentes articulen conocimientos, considerando los abordajes generados en contextos previos. Esta metodología facilitó la conexión entre los conceptos trabajados en diferentes etapas del proceso educativo, permitiendo que los docentes establecieran conexiones entre temas y brinden una perspectiva coherente y continua a los estudiantes. Esta capacidad para articular conocimientos mejora el proceso de enseñanza y aprendizaje, favoreciendo la retención y la comprensión a largo plazo [13].

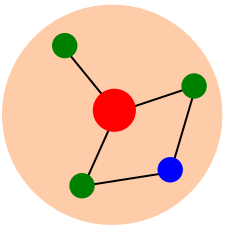


Figura2: Surgimiento de actividad inter-contextual

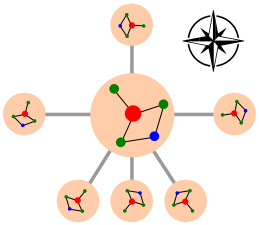


Figura3: Vinculación de nodos según la brújula de ideas

3.4 Navegando entre conceptos

Los conceptos se encuentran intrínsecamente relacionados en el contexto de una currícula de estudios, presentando similitudes, alternativas, antecedentes y consecuencias. Tal como se ha planteado en la investigación de Tseng, se ha propuesto una analogía simbólica, la “brújula de ideas”, que se puede emplear para explorar

sistemáticamente los conceptos del modelo. La ubicación espacial de los nodos puede apreciarse en la Figura 3.

La brújula de ideas representa una perspectiva novedosa y prometedora para la exploración intencionada de las conexiones conceptuales, ofreciendo una estructura visualmente representativa y organizada para comprender y analizar las interrelaciones entre los conceptos en el ´ámbito educativo. Al adoptar esta técnica se pudieron identificar patrones y descubrir nuevas asociaciones entre ideas.

En el marco de este modelo propuesto, se ha incorporado un apartado específico dentro del nodo conceptual que permitió establecer las conexiones cognitivas con otros nodos conceptuales, basándose en sus respectivas relaciones. Visualmente, esta estructura se organizó utilizando una topología análoga a los puntos cardinales, como se ha propuesto en el trabajo original de Tseng [16].

La identificación de conexiones y relaciones entre conceptos procedentes de distintas ´áreas del conocimiento brinda la oportunidad de integrar ideas y perspectivas diversas. Esta interdisciplinariedad enriquece la comprensión y el análisis del conocimiento, lo que facilita una visión más completa y holística de los temas abordados.

3.5 Diferentes tipos de nodos

Es necesario destacar que en este modelo, los nodos se clasifican en distintos tipos según su naturaleza. Los tipos de nodos identificados son los siguientes:

- **Nodos Conceptuales.** Estos nodos se encargan de alojar conceptos descontextualizados, sirviendo como referencias que se abordarán según diferentes contextos. Facilitan la navegación entre conceptos mediante la “brújula de ideas”.
- **Nodos Contextuales.** Estos nodos proporcionan una visión contextualizada del concepto central, refiriendo asimismo al contexto que los caracteriza.
- **Nodos de Actividad Inter-contextual.** Estos nodos contienen actividades, ideas, ejercicios o enfoques que permiten conectar conceptos caracterizados en dos o más contextos. Cada actividad, idea o ejercicio se representa mediante un nodo individual, pudiendo haber varios nodos enlazados a un mismo grupo de contextos.

La caracterización de estos nodos proporciona una distinción clara y sencilla en el modelo visual mediante los colores utilizados —rojo para nodos conceptuales, verde para nodos contextuales y azul para nodos de actividad intercontextual—, como sugiere Burch. Además, la topología del grafo resultante refleja la semántica inherente a cada nodo.

4. Resultados

Tras el proceso de desarrollo se ha alcanzado un modelo repetible que reúne todos los elementos favorables para su crecimiento orgánico. La utilización de una herramienta que gestione enlaces entre fichas, como Obsidian [8], ha permitido obtener una estructura desagregada, como puede verse en la Figura 4.

El modelo gen érico presentado dispone de los siguientes componentes:

1. Un meta-grafo con tres tipos de nodos.
2. Plantillas para cada tipo de nodo caracterizado.
   a. Ficha de concepto (definición, enlaces a contextos, brújula de ideas con enlaces)
   b. Ficha de contexto (abordaje)
   c. Ficha de actividad (objetivos de aprendizaje, duración, descripción de la actividad, competencias, evaluación)
3. Filtros y visualizaciones contextuales, facilitadas por la herramienta.

La construcción de este modelo es un esfuerzo colaborativo entre docentes de diversas asignaturas, permitiendo un trabajo asincrónico donde cada uno aporta desde su contexto para el crecimiento iterativo e incremental del grafo.

Este modelo facilita la navegación entre conceptos relacionados, permitiendo una exploración profunda de las aproximaciones contextuales a los conceptos. Asimismo, agiliza la generación de nuevas actividades inter-contextuales, brindándoles mayor sustento y pluralidad de abordajes. En consecuencia, el enfoque adoptado ha demostrado ser efectivo para construir un sistema de conocimiento interconectado y de crecimiento progresivo, brindando un enriquecimiento significativo a la comprensión y aplicación de los conceptos abordados.



Figura4: Mapeo entre modelo navegable y fichas descriptivas

5. Conclusiones

Este artículo presenta un enfoque innovador y flexible para abordar la interdisciplinariedad en la educación, presentando un modelo visual que permite la articulación de contenidos desde diferentes contextos. La subdivisión de los nodos de articulación en fichas específicas, junto con los enlaces correspondientes, ha permitido una mejor organización y comprensión de los conceptos enriquecidos por diversos enfoques. El meta-grafo, así organizado, permite establecer relaciones interdisciplinarias y generar objetos de aprendizaje que refuerzan la comprensión y aplicabilidad de los conocimientos adquiridos. Además, la "brújula de ideas" ha sido una herramienta valiosa para explorar sistemáticamente las conexiones conceptuales y enriquecer el análisis del conocimiento educativo. La identificación de diferentes tipos de nodos ha proporcionado una clasificación clara y efectiva en el modelo visual, lo

que ha permitido el filtrado de información, facilitando la comprensión y organización del mismo. En consecuencia, este enfoque ofrece una valiosa oportunidad para que docentes y estudiantes desarrollen una visión coherente y continua de los contenidos, promoviendo una educación más significativa y contextualizada.

Como trabajo futuro, se propone estudiar la repetibilidad y sistematización de este modelo, lo que abrirá nuevas perspectivas para contribuir a mejorar la calidad y eficacia de la educación en la actualidad, promoviendo la integración y aplicación de los conocimientos en distintos contextos y disciplinas.

6. Agradecimientos

Este trabajo se realiza en el contexto del proyecto: "Identificación de nodos potenciales de articulación en la carrera de ingeniería informática Universidad Nacional de La Matanza" (C2 ING 091).

La línea de investigación presentada colabora con la tesis de la Maestría en Tecnología Informática de Lucas Videla, dictada por la Universidad Abierta Interamericana.

Referencias

1. Ahrens, S.: How to Take Smart Notes: One Simple Technique to Boost Writing, Learning and Thinking: For Students, Academics and Nonfiction Book Writers. CreateSpace, North Charleston, SC (2017)
2. Alexander, C., Ishikawa, S., Silverstein, M.: A Pattern Language: Towns, Buildings, Construction. Oxford University Press, New York (1977)
3. Álvarez, R.P.: Formación superior basada en competencias, interdisciplinariedad y trabajo autónomo del estudiante. Revista iberoamericana de educación 35(1), 1–33 (2004)
4. Bjork, R.A.: On the symbiosis of remembering, forgetting, and learning. In: Successful Remembering and Successful Forgetting. Psychology Press (2010)
5. Burch, M., Huang, W., Wakefield, M., Purchase, H.C., Weiskopf, D., Hua, J.: The State of the Art in Empirical User Evaluation of Graph Visualizations. IEEE Access 9, 4173–4198 (2021). https://doi.org/10.1109/ACCESS.2020.3047616
6. Coplien, J.O.: Software Patterns. SIGS Management Briefings, SIGS, New York (1996)
7. Costa, V.A., Río, L.S.d.: La articulación en la enseñanza. In: I Jornadas sobre las Prácticas Docentes en la Universidad Pública. Transformaciones actuales y desafíos para los procesos de formación (La Plata, 2016) (2016)
8. DYNALIST INC.: Obsidian - sharpen your thinking., https://obsidian.md
9. Field, M., et al.: Assessing interdisciplinary learning. New directions for teaching and learning 58, 69–84 (1994)
10. Gamma, E. (ed.): Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley Professional Computing Series, Addison-Wesley, Reading, Mass (1995)
11. Giordano Lerena, R., Cirimelo, S., de Decanos de Ingeniería-CONFEDI, C.F.: Propuesta de estándares de segunda generación para la acreditación de carreras de ingeniería en la república argentina "libro rojo de confedi". Consejo Federal de Decanos de Ingeniería-CONFEDI. Universidad FASTA Ediciones (2018)
12. Jacobs, H.H.: Interdisciplinary curriculum: Design and implementation. ERIC (1989)

13. Kornell, N., Bjork, R.A.: Learning Concepts and Categories: Is Spacing the "Enemy of Induction"? Psychological Science 19(6), 585–592 (Jun 2008). https://doi.org/10.1111/j.1467-9280.2008.02127.x
14. Pérez, A., Soto, E., Sola, M., Serv´an, M.: Aprender cómo aprender. autonomía y responsabilidad: el aprendizaje de los estudiantes (2009)
15. Sweller, J.: Cognitive load during problem solving: Effects on learning. Cognitive science 12 (2), 257–285 (1988)
16. Tseng, F.L.: The essence of the Zettelkasten method, demystified. (Feb 2022), https://feeei.substack.com/p/the-essence-of-the-zettelkasten-method
17. Von Landesberger, T., Kuijper, A., Schreck, T., Kohlhammer, J., Van Wijk, J., Fekete, J.D., Fellner, D.: Visual Analysis of Large Graphs: State-of-the-Art and Future Research Challenges. Computer Graphics Forum 30(6), 1719–1749 (Sep 2011). https://doi.org/10.1111/j.1467-8659.2011.01898.x
18. Zhao, V., Tseng, F.L.: The Compass of Zettelkasten Thinking (May 2022), https://www.youtube.com/watch?v=-7r9t9T9Aww

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Aprendizaje Basado en Proyectos como propuesta de enseñanza para una asignatura introductoria de programación

Victoria Aragón and Gisela Dorzán

Departamento de Informática,
Facultad de Ciencias Físico Matemáticas y Naturales,
Universidad Nacional de San Luis
{vsaragon,mgdorzan}@unsl.edu.ar

Resumen El Aprendizaje Basado en Proyectos propone una alternativa a la enseñanza tradicional enfatizando la integración de contenidos y desarrollando habilidades requeridas para los futuros profesionales. En este artículo se formula una propuesta basada en dicha metodología para ser aplicada en la asignatura Introducción a la Programación de la carrera Licenciatura en Ciencias de la Computación de la Universidad Nacional de San Luis la cual consiste en el desarrollo de un proyecto, donde se realiza la definición del problema, incluyendo la elección de la realidad en la que se circunscribe dicho problema, el diseño y el desarrollo de un algoritmo que resuelve dicho problema implementado en un lenguaje de diseño. Se describen los tramos y etapas que conforman dicha propuesta, como así también la rúbrica utilizada para la evaluación. Los proyectos desarrollados en equipos de trabajo se difunden públicamente en las jornadas de promoción de carreras de la institución.

Keywords: Aprendizaje Basado en Proyectos, Programación, Propuesta de enseñanza

1. Introducción

Un tema de debate permanente en el contexto de las universidades trata sobre las acciones que deben ser llevadas a cabo para dar apoyo académico a los estudiantes para que logren ingresar, permanecer y egresar en tiempos razonables, asegurando un proceso de aprendizaje significativo continuo para su desarrollo. Por lo tanto, se debe buscar la manera de fortalecer e instalar políticas para el mejoramiento de la docencia universitaria, aportando a la formación pedagógica de los docentes. Se advierte con preocupación que, en muchas instituciones, incluida la Universidad Nacional de San Luis (UNSL), se sigue privilegiando la enseñanza sobre el aprendizaje, con métodos tradicionales centrados en la cátedra. Además, se prioriza lo memorístico y la reproducción de saberes por sobre la comprensión, donde los estudiantes tienden a asumir un papel pasivo en los procesos de enseñanza y aprendizaje.

En particular, se observa que el primer año universitario es altamente relevante en la configuración del destino académico y un tramo muy crítico por los altos ´índices de deserción, reprobación y, en general, resultados ineficientes al considerar aprendizajes significativos. Por lo tanto, es importante que los equipos docentes que se desempeñan en los primeros años de las carreras posean experiencia docente y un abanico de estrategias de posible aplicación en el aula, ya que es una de las variables institucionales

que más afecta el desempeño estudiantil. Dichas prácticas de enseñanza deben incluir procesos de revisión, análisis y reflexión, de manera que se retroalimenten desde la propia práctica.

Algunos docentes buscan alternativas que contribuyen a mejorar el aprendizaje de los estudiantes las cuales son pensadas y diseñadas con el convencimiento de que toda persona que lo desee y se esfuerce es capaz de comprender y que, a través de un proceso de enseñanza pertinente, se puede ayudar a que esto sea posible, favoreciendo la construcción del conocimiento. Se reconoce que, utilizando estrategias, diferentes a las tradicionales, se puede dar soporte a un mayor número de estudiantes quienes aprenden de diferentes maneras. El Aprendizaje Basado en Proyectos (ABP) es una base para promover en los estudiantes la construcción de comprensiones significativas en el transcurrir del tiempo, respondiendo a una de las problemáticas relativas a la Educación Superior, como es aportar a los procesos de enseñanza y de aprendizaje para favorecer y facilitar la comprensión de los estudiantes, mediante prácticas pedagógicas fundamentadas teóricamente.

Consideramos que es importante acercar al estudiante de la carrera Licenciatura en Ciencias de la Computación de la Facultad de Ciencias Físico Matemáticas y Naturales (FCFMyN) de la UNSL, desde el primer año, al perfil profesional a través de los contenidos curriculares de la asignatura Introducción a la Programación. En este sentido, se presenta una propuesta basada en ABP donde se pretende la realización de un proyecto para integrar los conceptos abordados en la asignatura y así acercarlos al perfil profesional que establece el Plan de Estudios.

El artículo se estructura de la siguiente manera: en la sección 2 y 3 se describen el marco teórico utilizado y los tramos del proyecto, respectivamente. Por último, se esbozan algunas conclusiones y trabajo a futuro.

2. Marco teórico

El Instituto Buck de Educación define al ABP como: “un método sistemático de enseñanza que involucra a los estudiantes en el aprendizaje de conocimientos y habilidades, a través de un proceso extendido de indagación, estructurado alrededor de preguntas complejas y auténticas, y tareas y productos cuidadosamente diseñados” [1].

El ABP es una metodología activa, multidisciplinar e integradora de aprendizaje fundamentada en el paradigma constructivista para explorar, conocer y comprender el mundo real. Estas actividades se llevan a cabo a través de la elaboración de proyectos en grupos de estudiantes, iniciando con una pregunta significativa, problema o desafío. Los estudiantes investigan de manera relativamente autónoma y con un alto nivel de implicación y colaboración para proponer una solución al problema, siendo actores activos, conscientes y responsables de su propio aprendizaje.

El ABP surge, por parte de Kilpatrick [2], pedagogo americano discípulo de John Dewey, como una concreción de los principios propuestos por la Escuela Nueva [3], los cuales destacan la relevancia de que los estudiantes puedan “aprender haciendo” considerando, además, los intereses de éstos como puntos de partida para el aprendizaje.

Diversos estudios científicos [4,5,6,7,8] comprobaron efectos positivos sobre el uso del ABP, entre los que se destacan: la satisfacción con el aprendizaje alcanzado y sentimiento de preparación para afrontar situaciones profesionales reales; la valoración positiva con respecto a los aprendizajes alcanzados; la mejora en la capacidad para trabajar en equipo con motivación e interés, de expresarse de forma oral, profundizar de conceptos y relacionarse con docentes y compañeros; la disminución del estrés en instancias evaluativas obteniendo mejores calificaciones; el aprendizaje de nuevos conceptos de la asignatura de forma fácil, amena e interesante y poder relacionarlos con temas transversales con otras asignaturas; la detección de errores en sus propias producciones; y el desarrollo de habilidades para la resolución de problemas, aprendizaje duradero y autónomo, que incluye diferentes puntos de vista evidenciando un razonamiento crítico, interacción social y metacognición, entre otros.

A pesar de los beneficios que reporta el ABP, investigaciones científicas [9,10,11] indican que existen dificultades para su aplicación, tanto para los estudiantes como para los profesores. Ambos grupos han descripto problemas en la gestión de los tiempos, en particular, los estudiantes reportaron inconvenientes al momento de precisar preguntas científicas, construir conocimiento y argumentar de forma lógica. Por otra parte, los profesores expresaron tensiones en relación a: la carga de trabajo docente para la puesta en marcha de los proyectos; la distribución horaria o disciplinar por materias o áreas de conocimiento; el manejo de la clase encontrando el equilibrio entre el aprendizaje autónomo y las intervenciones docentes y el control; la aparente imposibilidad de abordar todos los contenidos del currículum; las dificultades en la evaluación del proceso de adquisición de habilidades y destrezas; y la posibilidad de conflictos dentro del propio claustro a raíz del carácter innovador, entre otras.

3. Propuesta

A continuación, se presenta la propuesta de este trabajo que consiste en el desarrollo de un proyecto donde se realiza la definición del problema, que incluye la elección de la realidad en la cual se circunscribe dicho problema, el diseño y la implementación de un algoritmo en PsFlex [12]. El cronograma de aplicación es de 7 semanas, realizando entregas parciales incrementales ante compañeros y docentes, cada una con su correspondiente corrección informada.

La clase se divide en equipos de 2 estudiantes para desarrollar el proyecto, compartir experiencias y tomar decisiones. Los posibles roles son analista, programador y diseñador gráfico y pueden, los integrantes del equipo, cambiar su rol a lo largo del proyecto. El producto final es presentado en las jornadas de promoción de carreras de la FCFMyN a través de exposiciones. El propósito es que los estudiantes recorran las etapas involucradas en la resolución de un problema haciendo uso de la programación.

El diseño de esta propuesta se divide en tres tramos, cada uno compuesto por diversas etapas.

3.1 Tramo I: Planificación

Pregunta problematizadora: Los estudiantes generan una lista de preguntas que surgen durante la investigación, impulsando así la exploración y el descubrimiento, y finalmente seleccionan una pregunta problematizadora. La pregunta disparadora para generar un debate entre los estudiantes podría ser: ¿A qué me enfrentaré cuando me desempeñe como Licenciado en Ciencias de la Computación?

Formación de equipos de trabajo: El armado de equipos se realiza de acuerdo con la teoría Trabajo en Equipo-Logro Individual (TELI) [13], para el desarrollo en forma cooperativa del proyecto. La idea principal de TELI consiste en motivar a los estudiantes para que se animen mutuamente y se ayuden para alcanzar los objetivos de aprendizaje planteados. Se fomenta la participación activa de todos los miembros del equipo, estableciendo objetivos claros y específicos para cada uno de ellos, y se valora tanto el logro individual como el éxito del equipo en conjunto. Si un equipo quiere ganar recompensas, debe ayudar a sus integrantes a aprender. Para formar los equipos, se los divide en dos grupos, con y sin experiencia previa en programación. Cada equipo de trabajo debe tener dos integrantes, si es posible, uno de cada grupo.

Interdisciplinaridad: Se promueve una enseñanza que considera las especificidades de la carrera, pero al mismo tiempo fomenta la interdisciplinaridad entre diferentes áreas de interés. Aquí se consideran Programación, Matemática y la interacción con el medio para identificar problemas que podrían informatizarse para utilizar como realidad del proyecto.

Objetivos, Metas y Habilidades para el siglo XXI: Considerando que los objetivos representan propósitos a alcanzar en un tiempo determinado, mientras que las metas son objetivos específicos y medibles que contribuyen a la consecución de uno o más de los objetivos, para cada área de interés del proyecto se establecen los siguientes objetivos, metas y habilidades específicas.

Dentro de los objetivos de aprendizaje del área de Programación se consideran *plantear* el problema a resolver y *aplicar* de manera adecuada la metodología de trabajo para la resolución de los problemas computacionales vista en la asignatura. Considerando los objetivos de aprendizaje del área de Matemática se tiene en cuenta *formular* expresiones algebraicas y *realizar* cálculos matemáticos para obtener resultados. Con respecto a los objetivos de aprendizaje perseguidos en la interacción con el medio se considera *conocer* experiencias reales de profesionales de la disciplina relacionadas con las expectativas que tenían y la realidad vivida, y cómo los profesionales se enfrentaron a diferentes dificultades y las superaron. Además, se contempla la interacción con la comunidad para *desarrollar* capacidades comunicativas con entornos no informáticos y descubrir necesidades de la comunidad que puedan informatizarse.

Las metas planteadas para el área de Programación son: (1) al menos el 60% de los equipos enuncia textualmente un problema a resolver, con origen en la comunidad, haciendo uso de los conceptos dados en la asignatura; (2) a lo sumo el 40% de los equipos enuncia textualmente un problema a resolver sin haber interactuado exitosamente con la comunidad, haciendo uso de los conceptos dados en la asignatura; (3) el 100% de los equipos atraviesa las etapas de resolución de problema: comprensión del problema (identificación de datos de entrada y salida), concepción del plan (desagregación de tareas considerando legibilidad y reuso para resolver el problema),

codificación (uso correcto de la semántica y sintaxis de PsFlex) y ejecución (diseño de conjuntos de prueba que consideren diferentes casos representativos).

Para el área de Matemática se plantea la meta que, al menos, el 80% de los equipos aplica correctamente las fórmulas para calcular promedios, porcentajes, entre otras.

Considerando la interacción con el medio se plantean las siguientes metas: (1) al menos el 80% de los equipos asiste a las charlas con los profesionales que expongan; (2) al menos el 50% de los equipos formula preguntas a los profesionales; (3) al menos el 60% de los equipos obtiene datos suficientes de la comunidad para enunciar un problema a resolver computacionalmente.

En el Plan de Estudios OCD N °1/23 [14], que describe el perfil profesional del Licenciado en Ciencias de la Computación de la UNSL, se establece que el graduado "... cuenta con una formación integral que le permite el desempeño en equipos de trabajo, la comunicación efectiva y el aprendizaje continuo, además de desarrollarse con una actitud emprendedora, ética, responsable y de superación constante, consciente de los problemas sociales relacionados con el ejercicio de su profesión en el contexto local y global". Estas habilidades forman parte de las planteadas por Mariana Maggio en [15] quien las agrupa en tres categorías:

- Habilidades de aprendizaje e innovación: preparan para enfrentar la complejidad de la vida y el trabajo.
- Habilidades vinculadas a la información, los medios y la tecnología: son propias del entorno actual marcado por la tecnología y el acceso a abundante información.
- Habilidades para la vida y la carrera: permiten desenvolverse en ambientes complejos.

En la presente propuesta se pretende abordar, dentro de las *habilidades de aprendizaje e innovación*, la creatividad, el pensamiento crítico y el trabajo grupal cooperativo. Considerando las *habilidades vinculadas a la información* se plantea abordar la alfabetización digital (utilización de PsFlex, editor de textos, herramientas audiovisuales) y dentro de las *habilidades para la vida y la carrera* se consideran la flexibilidad y adaptabilidad, la autonomía, las habilidades sociales e interculturales, la productividad, la ética, el liderazgo, la responsabilidad y la tolerancia a la presión.

La meta planteada para las habilidades de aprendizaje e innovación es que el 100% de los equipos puede crear una solución al problema planteado minimizando los recursos necesarios y pueda explicar su génesis y el camino recorrido para hallar la solución. En cuanto a las habilidades vinculadas a la información se plantean las siguientes metas: (1) el 100% de los equipos puede codificar su plan para resolver el problema con PsFlex y (2) el 100% de los equipos utiliza herramientas informáticas para exponer el proyecto realizado. Finalmente, para las habilidades para la vida y la carrera se plantea la meta que el 100% de los equipos desarrollen el proyecto (resolución de un problema por ellos definido) en grupo sin plagio, respetando las fechas de entregas, resolviendo situaciones inesperadas a través de la toma de decisiones y sin arriesgar de modo directo la salud, la seguridad de la información conforme la ley, los derechos, los bienes o la formación de los habitantes.

3.2 Tramo II: Desarrollo e implementación

Recursos necesarios: Equipo docente de Introducción a la Programación, computadoras con conexión a internet, proyector, editor de texto y de presentación y sala para las presentaciones y reuniones con los profesionales.

Entregas parciales incrementales: La idea es que los estudiantes aborden los contenidos mínimos, los comprendan a través de los trabajos prácticos y los apliquen en el proyecto de manera incremental. En la primera entrega del proyecto se aplican los contenidos mínimos: Tipos de datos, operaciones y expresiones, Entrada y salida de datos, y Estructuras de control condicional. En la segunda entrega se incorporan Estructuras de control y Estructuras de Datos y, finalmente, en la tercera entrega, se agrega el contenido Subalgoritmos. En cada entrega se respeta la metodología de desarrollo de algoritmos: análisis, diseño, implementación y prueba.

Actividades pautadas según calendario y cronograma:

- Semana 1: Pregunta problematizadora. Charla con profesionales y la comunidad. Formación de equipos de trabajo y asignación de roles. Definición de realidades para proyecto.
- Semana 2-4: 3 Entregas parciales incrementales. Muestra a compañeros y docentes. Retroalimentación.
- Semana 5: Preparación para la muestra en jornadas de promoción de carreras.
- Semana 6: Muestra en jornadas de promoción de carreras. Semana 7: Reflexión sobre lo desarrollado y la exposición.

3.3 Tramo III: Presentación y evaluación del Producto Final

Difusión y Producto final: Desde el año 2002, se realizan jornadas de promoción de carreras, destinadas a estudiantes de los últimos años de las escuelas secundarias de toda la provincia, para que conozcan la propuesta de las diferentes carreras que ofrece la UNSL, con el propósito de profundizar la información sobre el perfil y el campo ocupacional de dichas carreras y colaborar con la orientación vocacional de los potenciales aspirantes. Por lo tanto, toma relevancia que los diferentes equipos de estudiantes realicen la exposición de sus proyectos a la comunidad en estas jornadas aportando al desarrollo de nuevas propuestas de comunicación.

Evaluación: Anijovich [16] define la evaluación formativa como “un proceso en el que se recaba información con el fin de revisar y modificar la enseñanza y el aprendizaje en función de las necesidades de los alumnos y expectativas de logro para alcanzar”. Teniendo en cuenta el tipo de metodología a la que pertenece el ABP, se considera que este tipo de evaluación es el más pertinente para ser aplicado. En este sentido, se pretende recolectar información durante el proceso de aprendizaje, utilizando bitácoras, de manera que quede expresado el modelo activo de aprendizaje, es decir, registrar la forma en la que van aprendiendo y así descubrir el modelo de distribución de conocimiento que se va efectivizando en el recorrido. Además, cada entrega parcial tiene una corrección informada por parte del equipo docente, cuyos criterios de evaluación se presentan en una rúbrica (ver Tabla 1), tanto para el estudiante (autoevaluación) como para el docente (heteroevaluación). También se realiza una evaluación sumativa a través de la presentación del proyecto desarrollado de acuerdo con el criterio “Exposición” de la rúbrica presentada.

Criterios	Puede mejorar	Satisfactorio	Excelente
Abstracción de la realidad	Características que no pueden ser abordadas con los CC, múltiples ambigüedades y datos de entrada/salida poco claros o incompletos.	El enunciado del problema, creado artificialmente, puede ser resuelto con los CC. Presenta ambigüedades, pero los datos de entrada/salida son claros.	Indaga diferentes necesidades en la comunidad y enuncia un problema que puede ser resuelto usando los CC. No presenta ambigüedades y los datos de entrada/salida son claros.
Concepción del plan	Las tareas no permiten llegar al resultado deseado. Las fórmulas matemáticas mencionadas no son las pertinentes al problema.	Las tareas permiten llegar al resultado deseado usando la metodología de desarrollo sin describirlas en profundidad empleando fórmulas matemáticas pertinentes.	Desarrolla y describe en profundidad las tareas que permiten llegar al resultado deseado usando la metodología de desarrollo, empleando fórmulas matemáticas pertinentes.
Codificación	El algoritmo no se condice con las tareas presentadas. Los objetos utilizados son inadecuados y/o insuficientes. No se aplican de manera correcta los CC de la entrega. No respeta los buenos hábitos de programación ni la sintaxis.	El algoritmo se condice parcialmente con las tareas presentadas. Se aplican correctamente los CC de la entrega. Define las variables necesarias. Solicita todos los datos de entrada necesarios. Respeta los buenos hábitos de programación y la sintaxis.	El algoritmo se condice con las tareas presentadas. Se aplican correctamente los CC de la entrega. Define las variables necesarias para almacenar los datos. Solicita todos los datos de entrada necesarios. Aplica los buenos hábitos de programación y la sintaxis.
Ejecución	No presenta datos de prueba ni su ejecución.	Presenta un conjunto de prueba con su ejecución.	Presenta al menos tres conjuntos de prueba representativos con su ejecución.
Interacción Con profesionales	No asiste a las charlas.	Escucha las exposiciones, pero no realiza consultas.	Escucha las exposiciones y realiza consultas pertinentes.
Exposición (evaluación de manera grupal e individual)	Ideas simplistas, no desarrolladas. Información mal estructurada y difícil de entender. Vocabulario muy limitado. Desarrollo de lo expuesto de manera	Demuestra conocimiento básico sobre el problema y su solución. Ocasionalmente habla en voz baja, comete algunos errores de pronunciación y con vocabulario limitado,	Demuestra un conocimiento completo sobre el problema y su solución. Articulación y pronunciación claras. Lenguaje apropiado y preciso. Mantiene contacto visual con la audiencia

	confusa y sin lógica aparente.	recurriendo a términos imprecisos. Mantiene contacto visual con la audiencia con interrupciones. Información presentada de manera lógica y coherente.	involucrándola en la presentación. Información presentada de manera lógica, coherente y creativa de forma que la audiencia lo puede seguir fácilmente.
Trabajo en equipo	Inexistente división de tareas. Poca interacción.	División igualitaria de tareas. Falta de resolución en tiempo y forma de lo asignado por parte de algún miembro. Pobre interacción.	División igualitaria de tareas. Interacción acertada y oportuna. Compromiso y responsabilidad.

Tabla 1: Rúbrica. CC: contenidos mínimos abordados en la asignatura.

Reflexión: Luego de la participación en las jornadas de promoción de carreras los estudiantes y docentes se reúnen para compartir sus experiencias en relación a las bitácoras de aprendizaje, las retroalimentaciones y la exposición a la comunidad de las producciones.

4. Conclusiones

Capacitarse en nuevas metodologías de enseñanza hoy es una necesidad más que una elección personal para los profesores universitarios. Es preciso que éstos redefinan sus concepciones arraigadas acerca de la enseñanza y el aprendizaje para poder dar respuesta a las exigencias planteadas a los futuros profesionales en la actualidad.

En este trabajo se presenta una propuesta basada en ABP para ingresantes a una carrera informática en la que se pretende integrar los conceptos abordados en la asignatura desarrollando habilidades requeridas por el Plan de Estudios con el objetivo de acercar a los estudiantes al perfil profesional. La propuesta incluye contacto con la comunidad para definir una problemática a ser abordada a través de los contenidos curriculares mínimos de la asignatura, desarrollar una solución en forma cooperativa, presentar la solución a la comunidad a través de las jornadas de promoción de carreras y, por último, reflexionar, usando como insumo las distintas instancias evaluativas, sobre el recorrido realizado para llegar a esta exposición.

Se considera que el ABP aquí presentado permite a los estudiantes desarrollar las habilidades de aprendizaje e innovación seleccionadas ya que el proyecto requiere que sean creativos, tanto para encontrar un problema como para exponer el producto final de forma tal que distintos públicos lo puedan comprender; y sean críticos, ya que deben seleccionar realidades pertinentes para ser abstraídas y abordadas con los contenidos curriculares de la asignatura desarrollando el proyecto con un par. El proyecto también implica el uso de distintos softwares, lo que contribuye con la alfabetización digital y también demanda el desarrollo de las habilidades escogidas para la vida y la carrera ya que los estudiantes deben vincularse con pares, profesores, profesionales de la

disciplina, la comunidad en general, considerando las correspondientes particularidades.

Se pretende que, al llevar a la práctica esta propuesta, se construya aprendizaje centrado en los estudiantes apoyando el pensamiento, la indagación y la metacognición, y se prepare para la vida elaborando una construcción de sentido en el proceso llevado a cabo. Aquí el docente debe cumplir un rol de facilitador que trata de hacer aflorar los puntos de vista de sus estudiantes y por consiguiente los valora, habilitando puentes que lleven desde la comprensión previa a nuevas concepciones más complejas. Como trabajo futuro se pretende aplicar esta propuesta en próximos dictados de la asignatura, documentando su desarrollo y así retroalimentar el proceso de mejora del ABP propuesto.

Referencias

1. Thomas J, Mergendoller J. and Michaelson A. Project-based learning: A handbookfor middle and high school teachers (1999)
2. Kilpatrick W. The Project Method. Columbia University's Teachers College Record.vol. XIX, num. 4 (1918)
3. Dewey J. Democracy and education: An introduction to the philosophy of education.New York: The Macmillan company (1916)
4. Gallagher S., Stepien W. and Roshenthal H. The effects of problem-based learningon problem solving. Gifted Child Quarterly, 36, 195-200. (1992)
5. Willard K. and Duffrin M. Utilizing project-based learning and competition to develop student skills and interest in producing quality food items. Journal of Food Science Education, 2, 69-73 (2003)
6. Restrego, B. Aprendizaje basado en problemas: una innovación didáctica para la enseñanza universitaria. Educación y Educadores, 8, 9-19. (2005).
7. Mioduser D. and Betzer N. The contribution of project-based learning to high achievers' acquisition of technological knowledge. International Journal of Techonology and Design Education, 18, 59-77. (2007).
8. Rodríguez-Sandoval E., Vargas-Solano E. and Luna-Cortés J. Evaluación de la estrategia. aprendizaje basado en proyectos". Educacio´n y educadores, 13(1), 13-25. (2010)
9. Marx R., Blumenfeld P., Krajcik J. and Soloway E. Enacting project-based sciences: Challenges for practices and policy. Elementary School Journal, 94, 517-538. (1997)
10. Krajcik J., Blumendfeld P., Marx R., Bass K., Fredericks J. and Soloway E. Inquiryin project-based science classrooms: Initial attempts by middle school students. The Journal of the Learning Sciences, 7, 313-350. (1998)
11. Van den Berg V., Mortermans D., Spooren P., Van Petegem P., Gijbels D. andVanthournout G. New assesment modes within project-based education the stakeholders. Studies in Educational Evaluation, 32, 345-368. (2006)
12. PsFlex. http://psflex.forjota.com.ar/
13. Slavin R. Aprendizaje Cooperativo: Teoría, Investigación y Práctica. Aique (2002) 14. Plan de Estudios OCD N ° 1/23 de la Licenciatura en Ciencias de la Computación. Universidad Nacional de San Luis. http://digesto.unsl.edu.ar/docs/202302/20230216193539_2976.pdf
14. Maggio M. Habilidades del siglo XXI: cuando el futuro es hoy: documento básico, XIII Foro Latinoamericano de Educación - 1a ed. - Ciudad Auto 'noma de Buenos Aires, Santillana (2018)
15. Anijovich R. and González C. Evaluar para aprender: conceptos e instrumentos. Buenos Aires: Aique Grupo Editor (2011)

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Enseñanza del Diseño de Aplicaciones Web mediante el Proceso Iconix Adaptado el Framework Odoo

Alejandro Sartorio[1,2], Matias Banega[1]

Universidad Abierta Interamericana. Facultad de Tecnología informática.
Centro de Altos Estudios en Tecnología Informática. Rosario, Argentina.
[alejandro.sartorio,matias.banega]@uai.edu.ar

Resumen Este artículo presenta una metodología original para la enseñanza del diseño de aplicaciones web full stack basadas en el framework Odoo Foundation, en su versión comunitaria de código abierto. La metodología se basa en el uso de recursos provistos por Odoo, como los HTML virtuales para la especificación de componentes de la capa Vista, las API para la representación de modelos y operaciones de control, y configuraciones basadas en el concepto de integración modular *Addons*. El nuevo proceso extiende los diagramas de Dominio y Robustez del proceso de diseño de software ICONIX permitiendo una nueva representación, a nivel de diseño, de las declaraciones y configuraciones que provee el framework Odoo. Estos diagramas ampliados formarán parte de una nueva etapa de especificación de componentes que se integrarán con las etapas preestablecidas de requerimientos, diseño e implementación de ICONIX.

Se presenta un caso de estudio de creación de una aplicación de gestión web aplicando este nuevo proceso y resaltando sus principales características. Esto permite realizar un análisis final de las ventajas sobre las mejoras de calidad, representatividad, automatización y conexión más explícita entre el diseño y la implementación del código.

Keywords: Metodologías de Software, Diseño y Especificación, Educación en Ingeniería de Software

1. Introducción

En las carreras informáticas, el diseño de software es una temática abordada en las etapas medias y finales. Generalmente, en las currículas académicas se comienza por el dictado de materias básicas referidas al pensamiento analítico, conceptos de abstracción y aplicación de metodologías para la resolución de problemas. Luego, se continúa con lenguajes de programación basados en los conceptos de estructuras de datos, seguido de cuestiones básicas de diseño y arquitectura de software. En las últimas etapas se consolidan conceptos, metodologías y herramientas de la ingeniería de software utilizada en el ciclo de vida de construcción de un software como ingeniería de requerimientos, metodologías de trabajo, arquitectura, diseño, herramientas y tecnologías, implementación y testing.

Por ejemplo, en la carrera de Tecnología Informática de la Universidad Abierta Americana (ISI-UAI) [1], las materias de Seminario de Aplicación Profesional y Trabajo Final de Ingeniería requieren que los alumnos realicen un trabajo práctico final que

[1] https://uai.edu.ar/facultades/

consiste en el desarrollo de una aplicación de gestión con tecnología Web de tres capas. El producto debe ser presentado con documentación de diseño y configuración. Particularmente, se utiliza la metodología de desarrollo ICONIX para la documentación de requerimientos, diseños de la arquitectura y solución de las reglas de negocios. Para la implementación se utiliza el framework .NET como primera opción y, para casos particulares, otros frameworks de características similares, como pueden ser Odoo Fundation en su versión comunitaria.

Otro ejemplo es la carrera de Licenciatura en Ciencias de la Computación de la Universidad Nacional de Rosario (LCC-UNR)[2], donde las materias Ingeniería de Software I y II incorporan especificaciones formales para la representación funcional de sistemas, cuestiones de diseño, arquitectura y testing.

En los ejemplos mencionados se comparten la mayoría de temas curriculares, con diferentes grados de profundidad y enfoques. La LCC-UNR profundiza en cuestiones de especificaciones formales, mientras que la ISI-UAI hace foco en herramientas de desarrollo de aplicaciones web en las tres capas de vista-modelo-controlador. Estos dos conceptos son importantes en la formación de profesionales para la industria del software[1,2,3,4,5]. Por este motivo, tener una formación base adecuada a los estándares actuales tendrá un doble impacto positivo para la academia, formación e industria. Por un lado, la calidad de los resultados de aprendizajes y competencias brindadas y adquiridas. Por otro lado, la mejor preparación de los recursos humanos, productos y herramientas que se incorporarán en la industria.

Este trabajo presenta una propuesta de enseñanza y aplicación de técnicas de especificación mediante el uso de una herramienta y metodología que se puede ajustar a los recursos pedagógicos adecuados que contempla los diversos criterios de desarrollo curricular implícitos en las definiciones de las diferentes carreras del área de informática. Esta propuesta se basa en las experiencias en docencia, investigación y transferencia transitadas durante los últimos años, en las que se combinan la metodología ICONIX, con los principios teóricos de la ingeniería de software, conceptos del área de la especificación formal para representar cuestiones funcionales y un framework orientado al desarrollo de aplicaciones web de gestión con arquitectura de tres capas del estilo MVC.

De esta manera, se combinan las propiedades del diseño y de las especificaciones funcionales a través de la extensión de los diagramas ICONIX para el Diagrama del Dominio y el Diagrama de Robustez, de manera que permita transferirlos de manera directa para su implementación en el framework de desarrollo Odoo.

La **especificación** y el **diseño** son dos conceptos distintos pero relacionados en el proceso de desarrollo de software [4]. La especificación se refiere a la descripción detallada de los requisitos, funcionalidades y comportamiento del software que se va a desarrollar [6]. En otras palabras, establece lo que el software debe hacer y cómo debe hacerlo. La especificación puede incluir diagramas, modelos, lenguajes formales, prototipos y otros artefactos que ayuden a visualizar y comunicar los requisitos y funcionalidades del software.

Por otro lado, el diseño se refiere a la creación de la arquitectura, componentes, módulos, interfaces y detalles técnicos del software que se va a desarrollar [7]. El diseño

[2] https://dcc.fceia.unr.edu.ar/es/lcc/materias

establece cómo se va a construir el software para cumplir con los requisitos y funcionalidades especificados. Puede incluir diagramas de clases, diagramas de secuencia, diagramas de componentes, diagramas de despliegue, entre otros artefactos que ayuden a visualizar y comunicar la estructura y los detalles técnicos del software.

En resumen, la especificación se enfoca en definir qué se debe hacer, mientras que el diseño se establece definir cómo se debe hacer. Ambos son importantes en el proceso de desarrollo de software y están interrelacionados, ya que el diseño se basa en la especificación y la especificación puede ser modificada en función del diseño [8]. El desafío de este trabajo es integrar ambos conceptos, a través de un recurso conocido en la Ingeniería de Software denominado ”desing driver”[9] que permite hacer representaciones de modelos genéricos, a partir de los requerimientos, y luego ir refinando y generando nuevos modelos que modelan características específicas.

Tras esta introducción, se continúa con una breve referencia del estado del arte sobre la enseñanza del diseño y la especificación aplicada a la construcción de software. Luego se plantea la propuesta de extensión de ICONIX para la incorporación de las propiedades de especificación funcional. Se continúa con un ejemplo de aplicación práctica. Por último, se presentan las conclusiones.

2. Estado del Arte

En esta sección, se identifican algunos de los conceptos y avances más importantes que han influido en el desarrollo de la metodología de especificación y diseño de software para aplicaciones web full stack.

En primer lugar, es importante destacar la importancia de la ingeniería de requisitos en el proceso de desarrollo de software [10]. La ingeniería de requisitos se enfoca en identificar, analizar y documentar los requisitos del software, lo que permite a los desarrolladores entender las necesidades de los usuarios y diseñar soluciones que satisfagan esas necesidades.

Otro avance importante fue la adopción de la arquitectura de software orientada a servicios (SOA) [11]. La SOA es una arquitectura de software que se basa en la idea de que las aplicaciones se pueden diseñar como una serie de servicios independientes que se comunican entre sí a través de interfaces estandarizadas.

La SOA ha sido especialmente importante en el diseño de aplicaciones web full stack, permitiendo una mayor modularidad y escalabilidad.

Además, la adopción de metodologías ágiles en la ingeniería de software ha sido fundamental en el desarrollo de instrumentos de especificación y diseño de software para aplicaciones web full stack [12]. A continuación, se presentan ejemplos de implementación y uso en las áreas de aplicación de interés para este trabajo.

2.1 Comparativa

Algunas de las características de las herramientas orientadas a la especificación [4] incluyen: a) Soporte para diferentes técnicas de modelado de requisitos, como modelos de casos de uso, modelos de objetos y modelos de estado. b) Soporte para la validación

de requisitos, lo que significa que la herramienta puede ayudar a identificar posibles errores en la formulación de los requerimientos. c) Soporte para la gestión de requisitos: la herramienta puede ayudar a rastrear los requisitos a lo largo del ciclo de vida del desarrollo de software.

Por otro lado, algunas de las herramientas basadas en el diseño [3,5] incluyen:

a) Soporte para diferentes modelos de diseño, como modelos arquitectónicos, modelos de componentes y modelos de datos. b) Soporte para la validación de diseño, lo que significa que la herramienta puede ayudar a identificar posibles errores en el diseño. c) Soporte para la gestión de diseño. La herramienta puede ayudar a rastrear el diseño a lo largo del ciclo de vida del desarrollo de software.

Para complementar, se muestra una tabla comparativa de las propiedades de la especificación y el diseño en función del propósito, características y beneficios.

Cuadro 1. Tabla comparativa

Característica	**Especificación**	**Diseño**
Propósito	Ayudar a especificar los requisitos de un sistema de software	Ayudar a diseñar un sistema de software
Características	Soporte para diferentes técnicas de modelado de requisitos, validación de requisitos y gestión de requisitos	Soporte para diferentes modelos de diseño, validación de diseño y gestión de diseño
Beneficios	Mejora la calidad del software al garantizar que los requisitos estén bien especificados y que el sistema esté bien diseñado	Mejora la calidad del software al garantizar que el sistema esté bien diseñado y que sea fácil de desarrollar y mantener

3. El modelo conceptual de ICONIX extendido

En esta sección comenzaremos con la descripción de los elementos principales que se tendrán en cuenta en las propuestas de la extensión de los diagramas ICONIX. En la figura 1 se visualizan tres capas de componentes diferentes. En primer lugar, están representados los elementos de la arquitectura de una aplicación web moderna estratificada en las capas Vista, Controlador, Modelo y API. En un segundo nivel se encuentran los diagramas ICONIX que se relacionan con los módulos de la arquitectura. Así, el diagrama de robustez es el encargado de modelar funcionalidades y artefactos de la capa Vista y Controlador. El diagrama de dominio interpreta el aspecto del controlador y el modelo de datos e integraciones con API integradas de forma nativa del framework de implementación.

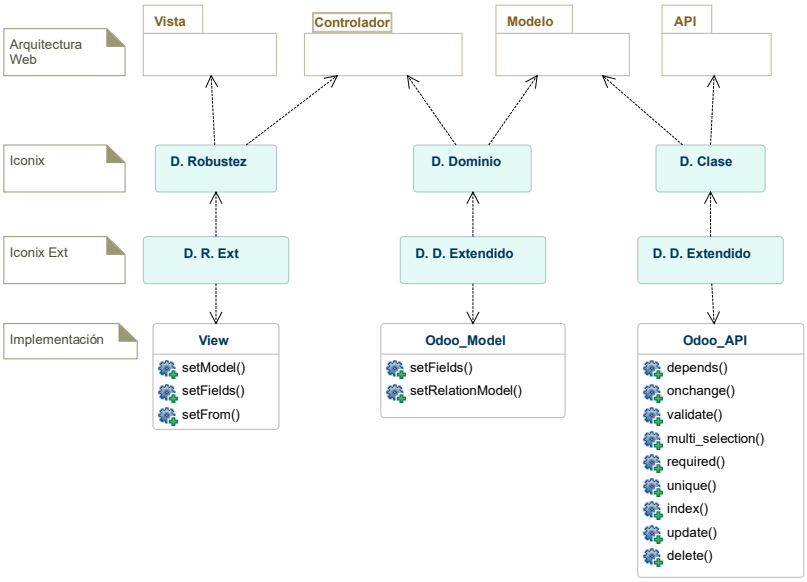


Figura1. El modelo conceptual ICONIX con extensiones

La tercera capa, denominada ICONIX Extendido, contiene los elementos que permitirán definir las especificaciones funcionales que se relacionarán con los componentes de la anterior capa (ICONIX). Para cada diagrama de ICONIX se le corresponde un lenguaje de especificación adecuado para relacionar de forma directa con las implementaciones en Odoo.

En la Tabla 2 se indican la correspondencia entre los diagramas de ICONIX y con las entidades de implementación definidas en Odoo para resolver las implementaciones.

La **fila 1** de la Tabla 2 indica que el Diagrama de Dominio (DD) se corresponde con un módulo de Odoo llamado ”Model”, utilizado para implementar los modelos de datos y comportamientos sobre los elementos del modelo. El lenguaje de especificación utilizado es Python[13], las entidades de DD se declaran como clases y las relaciones a través de un módulo nativo de Odoo, que se debe importar, denominado **fields**. En el fragmento de código 1.1 se puede observar las sintaxis para la creación de una clase (a través de la sentencia *class*), la referencia al nombre de la entidad (a través de la sentencia *_name=”nombre entidad”*) y las relaciones con otras entidades (a través de la sentencia *fields.Many2one(”nombre otra entidad”)*).

Cuadro 2. Tabla de relación entre ICONIX y Odoo

F	**Diagramas ICONIX**	→	**Componentes Odoo**
1	Dominio	→	Model
2	Prototipo	→	View → *Group, Sheet, Page, Form, Tree, Kamba, Menu*
3	Robustez	→	View → *Action* → *Model*
4	Secuencia	→	View → *Action* → *Model, Conrol*

5	Clase	→	Model, Control, View.action, View.menu Model → *API,* Control → *RPC* API → *api.returnapi.model, api.one, api.multi*

```
1   from odoo import fields ,  models , api
2   class NombreClase(models.Model :
3       _name = "nombre_entidad"
4       campo = fields.Many2one('nombre_entidad2')
```

Listing 1.1. Especificación del diagrama de dominio

La **fila 2** de la Tabla 2 hace referencia al Diagrama Prototipo de ICONIX en su relación con un módulo de Odoo llamado View, encargado de implementar los formularios con los que interactúa el usuario final de la aplicación. El View está compuesto por los elementos estándar HTML para la construcción de formularios para entrada y salida de información.

La **fila 3** de la Tabla 2 hace referencia al Diagrama de Robustez (DR) en su relación con los módulos de Odoo. Concretamente, los elementos del DR **Frontera**, **Acción** y **Entidad**; se relacionan respectivamente con los elementos View, Action y Model de Odoo. La sintaxis de la especificación ahora está representada por el lenguaje de marca XML[3] de la siguiente manera:

```
1   <record model="ir.ui.view" id="producto_form_view">
2       <field name="name">product.form</field>
3       <field name="model">nombre_del_modelo</field>
4       <field name="type">form</ field>
5       <field name="arch" type="xml">
6           <form string="Product">
7               <sheet>
8                   <group>
9                       <field name="campo del modelo"/>
10                  ...
```

Listing 1.2. Extensión del Diagrama de Robustez

Las dos primeras líneas del fragmento de código de especificación 1.2 son referencias a componentes y comportamientos de Odoo indicando el tipo de formulario que se utilizará.

La línea 3 hace referencia al modelo de datos que se relacionará con el formulario. Las líneas 4, 5, 6, 7 y 8 establecen elementos que tendrá el formulario. La línea 9 hace referencia al campo del modelo.

Siguiendo con la Tabla 2, la **fila 4** hace referencia a la relación del Diagrama de Secuencia con los elementos de **View**, **Action** y **Model** combinado con Control. En este caso, los elementos que intervienen son los mismos que se utilizaron en la fila 2, por lo cual, no es necesario agregar otras descripciones a las mencionadas en el fragmento de código 1.2.

[3] Extensible Markup Language (XML) - https://www.w3.org/TR/xml/

La **fila 5** de la Tabla 2 hace referencia a la relación del Diagrama de Clase (DC) con elementos de Odoo pertenecientes a la capa de Vista, Modelo y Controlador. En la capa Modelo se utiliza un módulo denominado API que permitirá extender funcionalidades para cada uno de los modelos implementados. Este recurso se implementa a través del concepto de decoradores de Python[14]. En el siguiente fragmento de código 1.3 se muestra un ejemplo de la aplicación de un decorador a una función encargada de restringir el valor de un dato ingresado para un campo:

```
1  @api.constrains('campo_nombre')
2  def funcion_a_implementar(self):
3  ...
```

Listing 1.3. Uso del módulo API como decorador

En la capa de **Controlador** se utilizan módulos, bibliotecas y otros componentes utilizados para implementar los algoritmos de control de una aplicación. Entre estos componentes se encuentran los de infraestructura relacionadas con: *Enrutadores*, *Callback*, *Integración entre controlador y vistas*, *ORM*, *Protocolos de acceso a datos*, *Templates* (planillas), *Internacionalización*, *Seguridad*, *Interoperabilida*, *HTML virtual*, Librerías de *Javascript*, Librerías de hojas de estilos *CSS*, Integraciones modulares tipos *addons*, etc.[15]

De esta manera, queda caracterizada conceptualmente la solución que se propone en este trabajo para la promoción de la extensión de ICONIX con especificaciones funcionales.

En particular, en los diagramas de dominio (DD) y robustez (DR), las especificaciones formales se pueden utilizar para especificar los requisitos del dominio de una manera precisa y concisa, asegurando que el sistema cumpla con las funcionalidades declaradas y que no contenga errores.

Por ejemplo, en el DD extendido, mediante la declaración de una función en Python decorada con el decorador *@api.constrains*, que contiene una sentencia de construcción de conjuntos por compresión sobre los valores posibles de un campo.

```
1  @api.constrains('employee_id')
2  def    _check_current_contract(self):
3  for contract in \
4  self.filtered(lambda c: \
5         (c.state not in ['draft' ,'cancel']):
```

Listing 1.4. "Fragmento del agregado de propiedades de un modelo de Odoo

Otro ejemplo: en el diagrama de robustez (DR) extendido, se incorpora una restricción de dominio para un determinado campo de un modelo. En el fragmento de código siguiente, en la línea 4, el campo *campo restringido* está restringido por el valor del campo *otro campo*.

```
<group>
    <field name="lista _componentes"">
        <tree editable="bottom">
            <field name="campo.restringido"
                   domain="[('otro.campo', '!=' , False)]"
            </field>
     ...
```

Listing 1.5. "Fragmento de la especifición de la vista de Odoo"

Las especificaciones formales también permiten verificar el comportamiento del sistema. Esto se puede hacer utilizando una variedad de técnicas, como la prueba formal, el modelado formal y la verificación formal.

Las especificaciones formales también facilitan la creación y el entendimiento de la documentación para programadores y actores interesados en el ciclo de vida del producto. Esto se debe a que las especificaciones formales son precisas y concisas, y proporcionan una visión clara del sistema.

En general, las especificaciones formales pueden ser una herramienta valiosa para ayudar a garantizar la calidad del sistema. Pueden ayudar a identificar y corregir errores, y pueden ayudar a garantizar que el sistema cumpla con los requisitos del usuario. Además, facilitan los procesos de prueba, validación del diseño de soluciones, etc.

4. Ejemplo de la aplicación de ICONIX extendido

En esta sección se muestra un ejemplo de un caso de uso definido para una aplicación del dominio de gestión de consultas médicas para una clínica médica. Por cuestiones de espacio y simplificación, se presenta un recorte de las características del caso de uso, del diagrama de dominio más las especificaciones y del diagrama de robustez más las especificaciones.

El caso de uso en cuestión tiene que ver con el registro de las consultas que se le asignan a los médicos de la clínica. El procedimiento se realiza a través de un formulario web donde el usuario debe ingresar los siguientes datos: médico, paciente, día, horario. Esa información debe quedar guardada y accesible para que el médico pueda agregar información referente a las prescripciones médicas que eventualmente se realicen. Los datos del paciente y el médico son extraídos desde un padrón que deberá estar previamente cargado y disponible para este propósito. El usuario podrá visualizar, a través de un formulario, las consultas previamente almacenadas con la posibilidad de filtrar, agrupar y acceder a cada uno de los registros.

El diagrama de dominio de la Figura 2 representa el modelo de almacenamiento de una consulta médica a través de la entidad denominada *hospital.appointment* que a su vez se relaciona con las entidades de pacientes (denominada *hospital.patient*), médicos (denominada *hospital.doctor*) y prescripciones (denominada *hospital.prescription*).

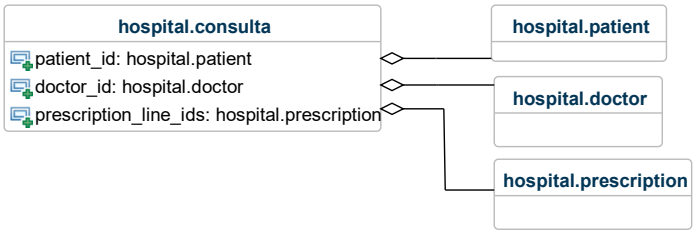


Figura2. Diagrama de Dominio ICONIX - CU: consulta médica

El fragmento de código en 1.6 se encuentra la especificación formal correspondiente al diagrama anterior.

```
1   class HospitalAppointment(models.Model):
2       .name = "hospital.appointment"
3       .description = "Hospital Appointment"
4
5       patient_id = fields.Many2one('hospital.patient')
6       doctor_id = fields. Many2one('hospital.doctor')
```

Listing 1.6. Especifición del diagrama de dominio

Ahora se continuá con la elaboración del diagrama de robustez. La Figura 3 identifica, en primer lugar, al elemento frontera llamado View, un componente nativo de Odoo que hereda de una clase que representa el formulario de carga de la consulta que interactuá con el usuario. Luego se encuentran las acciones conectadas a los formularios para confirmar, validar, guardar, editar, cancelar, seleccionar, etc. Las acciones pintadas con fondo de color son nativas de Odoo, por ese motivo no requieren implementación. Por último, se encuentran representadas las entidades que se ajustan a las definidas en el diagrama de dominio y sus especificaciones correspondientes.

Las especificaciones derivadas del diagrama de robustez se detallan en los fragmentos de código siguientes. En el Fragmento de Código 1.7, en la línea 1 se declara la frontera View, en la línea 2 se detalla la frontera Formulario, en las líneas 7-9 se mencionan las relaciones con las entidades declaradas en el diagrama de dominio. En la línea 10 se define una acción denominada *Confirmación* que deberá estar implementada por la capa de controladores y tendrá conexión con las entidades definidas anteriormente.

```
1   <record model="ir.ui.view" id="producto_form_view">
2       <field name="name">appoiment.form</field>
3       <field name="model">hospital.appoiment</field>
4       <field name="type">form</field>
```

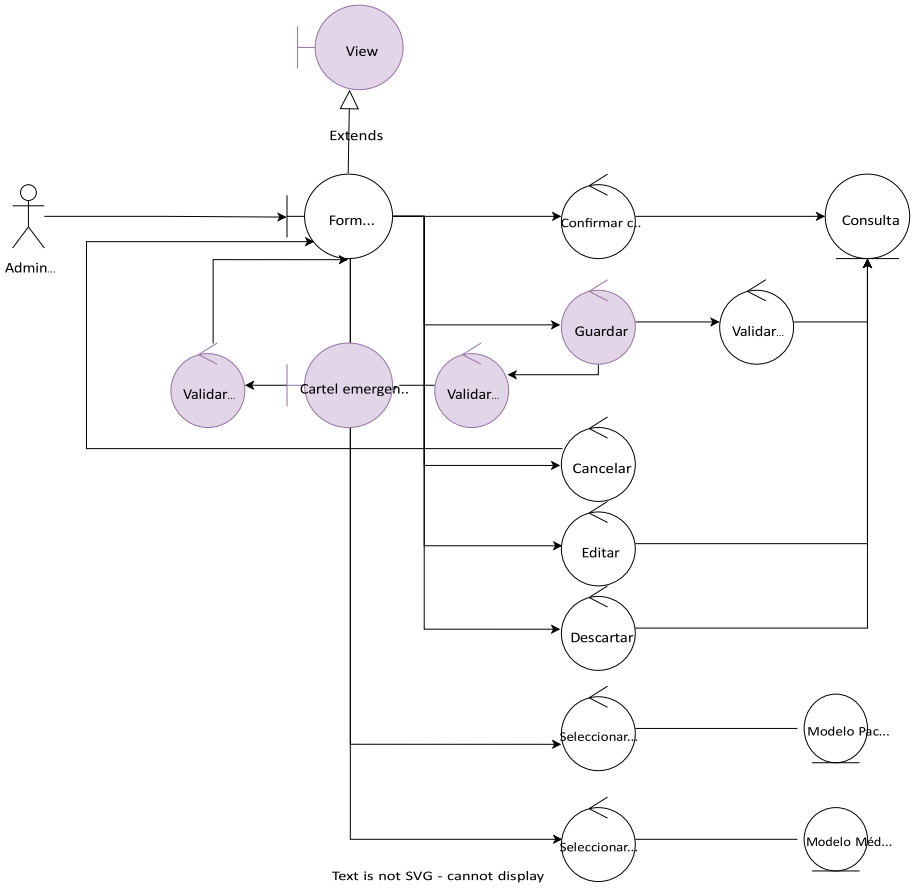


Figura3. Diagrama de Robustez ICONIX - CU: Consulta Hospital

```
5  <field name="patient_id" />
6  <field name="doctor_id" />
7  <field name="prescription_line_ids" />
8  <button type="object"
9          name="action_confirmar"
10         string="Confirmar" />
11 ...
```

Listing 1.7. Extensión del Diagrama de Robustez

5. Conclusiones

En este artículo, se propuso una hoja de ruta para introducir los conceptos de diseño y especificación en las carreras afines a la Informática. Esta propuesta se basa en el uso de metodologías de desarrollo de software actuales, como el modelo de desarrollo ágil.

Se creó un modelo conceptual que permite combinar el diseño con la implementación a través de especificaciones funcionales en los ecosistemas de desarrollo de aplicaciones web. Este modelo se basa en la idea de que las especificaciones funcionales son un lenguaje común que puede mejorar la conexión e integración entre el diseño y la implementación de aplicaciones de software.

También, se muestra que el modelo conceptual de extensión de la metodología ICONIX puede utilizarse para construir aplicaciones web que son independientes de la herramienta que se utiliza para la implementación. Esto se debe a que las especificaciones funcionales están escritas en un lenguaje estándar (Python y XML) que puede ser interpretado por cualquier herramienta de desarrollo.

De esta manera, al tener otro instrumento para introducir los conceptos de diseño y especificación tempranamente en las asignaturas afines a la Ingeniería de Software, se potencia el desarrollo de habilidades y competencias necesarias en la formación académica para construir software de calidad.

Referencias

1. Hussain, D., and Linda S.: Software engineering, bridging theory and practice in an agile learning environment. 2022 IEEE Global Engineering Education Conference (EDUCON). IEEE, 2022.
2. Ouhbi, S., Nuno P.: Software engineering education: Challenges and perspectives. 2020 IEEE Global Engineering Education Conference (EDUCON). IEEE, 2020.
3. GROENEVELD, W., VENNEKENS, J.; AERTS, K.: Software engineering education beyond the technical: A systematic literature review. arXiv preprint arXiv:1910.09865, 2019.
4. STRIUK, A., SEMERIKOV, S.: The dawn of software engineering education. 2019.
5. AL-RATROUT, S.: Impact of using agile methods in software engineering education: a case study. En 2019 6th International Conference on Control, Decision and Information Technologies (CoDIT). IEEE, 2019. p. 1986-1991.
6. Somé, S., Beucher, S., Huchard, M.: Software specification and design. A review of current approaches. En: IEEE 40th Annual Computer Software and Applications Conference (COMPSAC), Atlanta, GA, 2016, pp. 563-568.
7. Pressman, S.: Software Engineering: A Practitioner's Approach. En: 8th ed. New York: McGraw-Hill, 2014.
8. Glass, L.: Evolving software specification and design. En: Proceedings of the 5th international conference on Software engineering, San Diego, CA, 1981, pp. 2-9.
9. Latif, S. M., Rehman, A. S.,Khan, A.: The impact of design drivers on the performance of virtual reality applications. En: IEEE Transactions on Visualization and Computer Graphics,1999 26(12), 3016-3026.
10. Sommerville, I.: Ingeniería de software. En: (10ma ed.). Pearson Educación, 2016. 11. Erl, T.: Service-Oriented Architecture: Concepts, Technology, and Design. En: Prentice Hall, 2005
11. Beck, K., & Fowler, M.: Planning Extreme Programming. En: Addison-Wesley Professional, 2001
12. van Rossum, G.:. Python: Un lenguaje de programación para principiantes. The Python Journal, 1(1), 1-10., 2011.
13. DZICHKOVSKII, I.: Mecanismos reutilizables para mejorar la confiabilidad de aplicaciones desarrolladas con Python. 2020.

14. Pérez, J., López, P., Sánchez, M.: La infraestructura tecnológica de un framework de desarrollo web. En: IEEE Transactions on Software Engineering, 2023 49(1), 1-10. doi:10.1109/TSE.2023.3125682

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Programación por pares como estrategia de enseñanza en contextos de masividad

Gladys Dapozo, Cristina Greiner, Raquel Petris,
Ana María Company, María Cecilia Espíndola

Facultad de Ciencias Exactas y Naturales y Agrimensura, 9 de julio 1449 (3400) Corrientes
Universidad Nacional del Nordeste (UNNE)
gndapozo, cgreiner, rpetris, mcespindola, anamacom {@exa.unne.edu.ar}

Resumen. La enseñanza de la programación en la universidad, orientada específicamente a la formación de profesionales informáticos, se encuentra en un momento crítico debido a una importante demanda por la formación, lo que ha llevado a un notable incremento de la matrícula de las carreras de informática del país. La asignatura Algoritmos y Estructuras de Datos I, ubicada en el primer cuatrimestre del primer año de la Licenciatura en Sistemas de Información de la UNNE, para el dictado 2023 incorporó una adaptación de la programación por pares (pair programming) como estrategia de enseñanza para mitigar las dificultades de la masividad. Como resultado se destaca que los estudiantes han apreciado la modalidad, pero señalaron también cuestiones que deberían tenerse en cuenta en una próxima implementación de la metodología.

Palabras claves: Enseñanza de la Programación. Programación por pares (pair programming).

1. Introducción

La enseñanza de la programación en la universidad, orientada específicamente a la formación de profesionales informáticos, se encuentra en un momento crítico. Como consecuencia de la aceleración de la virtualización provocada por la pandemia, la masificación de la modalidad de trabajo remoto, la aparición de nuevas modalidades contractuales laborales que generan tentadoras oportunidades de facturación en dólares, ha llevado a un notable incremento de la matrícula de las carreras de informática.

La Licenciatura en Sistemas de Información de la Facultad de Ciencias Exactas y Naturales y Agrimensura (FaCENA) de la Universidad Nacional del Nordeste (UNNE) en el año 2023 ha tenido 1.128 inscriptos, superando ampliamente su promedio de 262 alumnos en los últimos 5 años (2018-2022)[*]. Dada esta gran demanda, la primera materia que aborda conceptos de programación - Algoritmos y Estructuras de Datos I - ubicada en el primer cuatrimestre del primer año, para este dictado incorporó la programación por pares como estrategia de enseñanza.

A continuación, se muestran las características de esta propuesta.

2. Características de la propuesta

[*] Informe de autoevaluación Acreditación CONEAU LSI-UNNE 2023.

La masividad de alumnos del ciclo lectivo 2023 exigió al equipo docente reorganizar el uso de los recursos, tanto materiales (aulas, equipamiento) como humanos (docentes), y optar por estrategias pedagógicas de autocontención y permanencia de los alumnos en las aulas.

El trabajo colaborativo es un proceso de interacción en el que los alumnos aprenden más de lo que aprendería en un contexto individual, la preguntas y repreguntas que con frecuencia ellos se hacen provocan respuestas - erróneas o no-, que dinamizan los contenidos conceptuales que ayudan a diferenciar, identificar y contrastar diferentes puntos de vista, de tal manera, que llegan a generar un proceso de construcción y acomodamiento de conocimiento [1].

En particular, se eligió la modalidad de programación por pares (*pair programing*) dado que es una metodología de desarrollo ágil que se utiliza en el mundo del trabajo. La programación por pares es un enfoque ágil en el que dos personas trabajan juntas de manera sistemática para desarrollar aplicaciones informáticas en tiempo reducido. Ha sido utilizada tanto en la industria del software como en la enseñanza de la programación y ha demostrado su efectividad en ambos escenarios [2]. Cada pareja debe adoptar e intercambiar los roles de conductor y navegador periódicamente. El conductor es la persona que utiliza el teclado en un momento determinado para escribir el código del programa. El navegador es la persona que retroalimenta al conductor a través de indicaciones, sugerencias y correcciones [3].

Para la implementación de la estrategia se solicitó a los alumnos de las 7 comisiones de aproximadamente 150 alumnos cada una, que gestionen e informen sus grupos de pares a través de un formulario digital.

Las clases de modalidad teórico-prácticas conllevan una serie de ejercicios que permiten comprender, reforzar y aplicar los contenidos conceptuales, y la correspondiente elaboración del código en lenguaje C que proponga una posible solución a los ejercicios planteados. Luego las soluciones de algunos pares de alumnos son expuestas para todos los asistentes, propiciando el debate y otras posibilidades o propuestas alternativas de soluciones. Si bien la metodología sostiene los roles de conductor y navegador, en esta propuesta no se puso el foco en estos roles, sino más bien en el trabajo colaborativo entre pares.

De este modo, se genera un espacio interactivo, de debate, que invita a los alumnos a pensar y analizar otras propuestas de solución, lo cual demanda conjugar esfuerzos, conceptos y competencias, mediante una serie de transacciones que les permita elaborar las soluciones de manera consensuada, tal como lo manifiesta [4] “Más que una técnica, el trabajo colaborativo es considerado una filosofía de interacción y una forma personal de trabajo, que implica el manejo de aspectos, tales como el respeto a las contribuciones individuales de los miembros del grupo”.

La primera instancia de evaluación parcial también se realizó en la modalidad por pares, dado que los primeros aprendizajes se llevaron a cabo de esta forma. Los resultados obtenidos han sido satisfactorios, considerando que además favoreció la permanencia de alumnos, disminuyendo el abandono prematuro.

Para la segunda etapa de aprendizaje, si bien se continuó trabajando en las clases en modalidad por pares, para las instancias evaluativas se volvió a la modalidad tradicional de evaluación individual.

Al finalizar el cursado se implementó una encuesta de respuesta voluntaria a los alumnos matriculados en el aula virtual de la asignatura.

3. Resultados

166 estudiantes respondieron la encuesta, y algunos de los resultados destacados son:

3.1 Comisión a la que pertenece:

La asignatura organizó las actividades en 7 comisiones a cargo de 2 docentes cada una. Se desarrollaron 3 clases teórico prácticas semanales (2 presenciales y 1 virtual sincrónica) de 2 horas de duración cada una.

Si bien las comisiones respetaron la planificación única de la asignatura, con la aplicación de la misma metodología y fechas de exámenes, los alumnos podrían tener percepciones diferentes de acuerdo al estilo de enseñanza de sus profesores. En la Fig. 1 puede apreciarse que en las respuestas están representadas todas las comisiones.

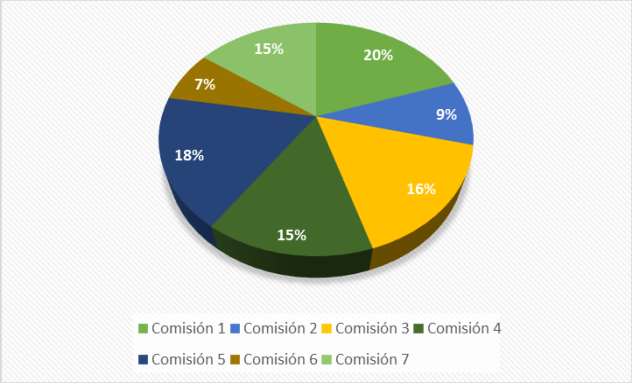


Fig.1. Alumnos por comisión

3.2 Género:

Una problemática vigente en la formación de informáticos es la baja participación de las mujeres. Si bien las políticas públicas están orientadas a fomentar vocaciones TIC en las jóvenes, es aún bajo este porcentaje. En la Fig. 2 se muestra la participación de las estudiantes en la encuesta.

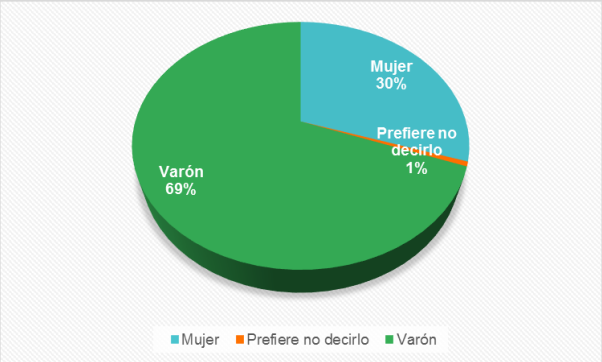


Fig.2. Alumnos por género

3.3 La modalidad de trabajar por pares le resulta:

Consultados acercar de su opinión sobre la modalidad de trabajo por pares, se les brindó una lista de opciones, pero podían agregar otras. La Fig. 3 permite apreciar que la mayoría indicó que le pareció buena o muy buena (83%).

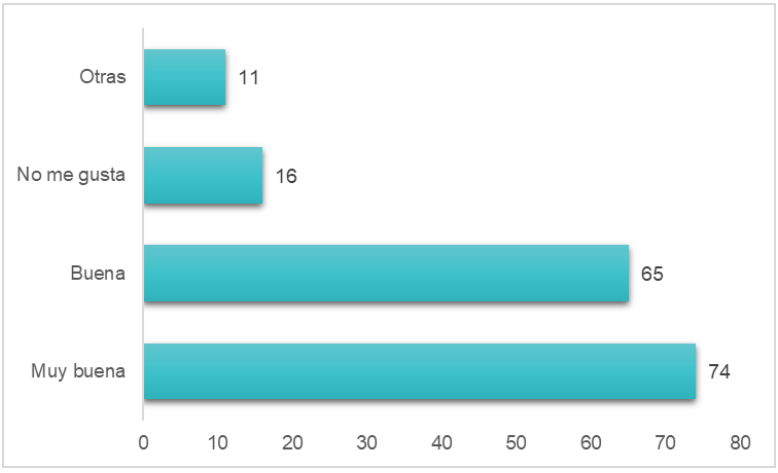


Fig.3. Opinión sobre la modalidad por pares

En la categoría **Otras**, 11 estudiantes expresaron lo siguiente:

1. *A mi parecer hace que le reste importancia a realizar mi trabajo y hace que lo tome más a la ligera.*
2. *Creo que podría funcionar de una buena forma, pero en el caso de que un compañero deje de cursar a mitad de cursar la materia, no resulta tan sencillo para los alumnos encontrar nuevos compañeros, al menos para los ingresantes, en otros años no sabría cómo puede resultar.*
3. *Es muy difícil ponerse de acuerdo*

4. *Lo hice solo todos los parciales*
5. *Me pareció muy buena, pero hubo algunos problemas sobre todo con gente cuyos compañeros abandonaron la facultad (no fue mí caso), pero creo que deberían ser más flexibles en esos casos para que el que se quede pueda seguir cursando correctamente.*
6. *Muy buena, pero, la implementación a largo plazo yo creo que puede ser mejor, siempre y cuando sean de a 2 nada más.*
7. *Muy Buena, cuando ambos integrantes han estudiado y practicado lo suficiente para contribuir de igual modo a la resolución de las tareas*
8. *No hice compañeros.*
9. *Personalmente prefiero la forma individual, pero reconozco sus beneficios específicamente al comienzo de la carrera, después me fue muy bien individualmente*
10. *Regular*
11. *Rendí solo. Por el trabajo no puedo coordinar horario de estudio*

En estas opiniones individuales, se destaca una cuestión que debería tenerse en cuenta en la metodología de implementación: las consideraciones necesarias para el caso de que alguno del par abandone el cursado, situación bastante frecuente en el transcurso de materias de primer año.

3.4 La modalidad de trabajar por pares le permitió:

Para consultar acerca de los beneficios de la modalidad por pares se les brindó una lista de opciones, pero podían agregar otras. La Fig. 4 permite apreciar que la mayoría destaca como positivo el debatir ideas y conceptos. Esto es particularmente interesante en la enseñanza de la programación dado que no existe una solución única para los problemas que se plantean en las guías de trabajos prácticos. Este debate entre pares contribuye significativamente en el aprendizaje.

Otro aspecto importante en la universidad, principalmente en el primer año, es tener un compañero de estudio. La modalidad por pares permitió que este acercamiento a otro, dado por condiciones académicas, facilitara el contacto para lograr un compañero de estudio.

Con esta modalidad se pretendía también facilitar el acceso al equipamiento necesario para la realización de los trabajos prácticos. Esta modalidad permitió que quienes no tenían una notebook pudieran realizar las actividades prácticas con los pares que sí contaban con el equipo.

Otro aspecto señalado fue la creación de hábitos de estudios, aspecto importante para el avance en la carrera. En alumnos ingresantes es bastante común que carezcan de metodologías de estudio que les permita organizar mejor su tiempo. Esta modalidad puede contribuir al desarrollo de hábitos de estudio.

En menor medida, los alumnos consideran que la modalidad por pares les permitió aclarar dudas sobre los prácticos, llevar la materia al día y adaptarse a la vida universitaria.



Fig.4. Lo que permitió la modalidad por pares

Los que optaron por Otras cuestiones, señalaron lo siguiente:

1. El trabajo por pares no fue muy diferente de lo habitual puesto que mi compañero de estudio y yo teníamos formas diferentes de afrontar los problemas por lo que hicimos la mayor parte de los trabajos individualmente.
2. Llevar la materia al día, compañero de estudio, debatir ideas y aclarar dudas (El alumno quiso abarcar tres opciones, no una sola).
3. Me permitió desarrollar la tolerancia y la forma de expresarme para poder transmitir lo que creo comprender de los conceptos teóricos.
4. Me resulta obligación. Mi compañera no me ayudaba porque estaba cursando otras materias y no se organizaba.
5. Mi compañero no hizo nada.
6. No conseguí compañero.
7. No la apliqué.
8. No la apliqué.
9. No me ayudó, porque trabajo y tengo cuestiones personales que no me permitían juntarme con mi compañera.
10. Si bien entiendo la importancia del trabajo en equipo dentro del ámbito de la programación, no me gusto tener que juntarme con una persona aleatoria que no estaba a mi nivel, esencialmente, para mí el primer parcial lo hice solo y alguien ligo nota gratis.
11. Tener la posibilidad de enseñar los temas adquiridos para reforzar mi conocimiento.

En estas opiniones individuales, se destacan situaciones en las que la modalidad no funcionó por las características de los pares, fuerte individualismo y escasa capacidad

de trabajo en equipo. Por otra parte, se destacan situaciones en las que el par reconoce ventajas en tener que enseñar a otro.

Los pares desbalanceados en cuanto a capacidades, suelen ser un problema. También aparece esta situación, pero no en un porcentaje significativo, lo que habla muy bien de los estudiantes que afrontaron la situación. Cabe aclarar también que los estudiantes que aportaron poco al equipo (par, en este caso), quedaron en evidencia en el segundo parcial dado que este fue individual.

3.5 Dificultades o inconvenientes de la modalidad:

Consultados acercar de las dificultades o inconvenientes que tuvieron en la modalidad por pares, se les brindó una lista de opciones, pero podían agregar otras. La Fig. 5 permite apreciar que la mayoría (61%) destaca como dificultades cuestiones de organización, como días, horarios y lugares para estudiar.

Por otra parte, 22 estudiantes (13%) indica como inconvenientes la falta de acuerdos para la resolución de los problemas y la inflexibilidad de alguna de las partes. Sobre estos aspectos, propios del trabajo en equipo, debería realizarse una capacitación previa hacia los estudiantes a fin de facilitar el cumplimiento de los objetivos de la modalidad.

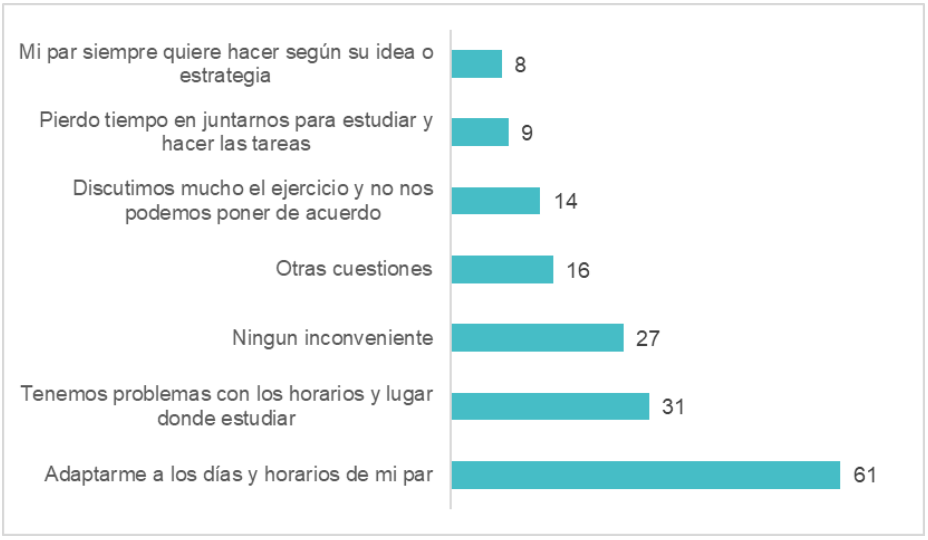


Fig.5. Inconvenientes de la modalidad por pares

28 estudiantes (16%) indicaron que no han tenido ningún inconveniente y 15 (10%) han indicado otras cuestiones, que se detallan a continuación:

1. *Tuve problema a la hora de resolver el problema*
2. *Trabaje solo en mi casa, solamente rendi el parcial con compañero.*
3. *Trabajar con un compañero que no estudió para el examen.*
4. *Te puede tocar alguno que sea un queso*

5. *Se cruzan las diferentes formas de resolver un problema de cada uno inconscientemente y puede generar varios problemas*
6. *Mi par no tenía el mismo interés en la materia*
7. *No la apliqué*
8. *Mi par no asistió a clases y termino perjudicándome en el primer parcial*
9. *Mi par no le dedica el tiempo suficiente a la materia porque trabaja*
10. *Mi par no estudia ni un poco*
11. *Mi par no colabora y termino trabajando individualmente (no me aporta nada)*
12. *Mi compañero no hizo nada*
13. *Estrategias o ideas distintas*
14. *Es una ilusión de trabajo grupal, en realidad uno solo termina haciendo el código*
15. *Con todo el debate, durante el parcial te puede llegar a faltar tiempo*

En estas opiniones individuales, se destacan situaciones en las que alguno de los pares no ha estado a la altura de lo esperado o no se ha organizado para poder contribuir en el equipo.

4. Conclusiones y futuros trabajos

La modalidad por pares utilizada en este contexto de masividad puede constituirse en una valiosa estrategia para la enseñanza de la programación.

Los estudiantes, en general, la reconocen como una buena estrategia y han destacado que les ha permitido el debate e intercambio de ideas, tener un compañero de estudio y adquirir hábitos de estudio.

Desde la asignatura se reconoce que ha permitido disminuir el volumen de exámenes para corregir, favoreciendo el cumplimiento de la planificación de las actividades. Así también la modalidad es válida para resolver la disponibilidad de equipamiento. En contextos de masividad es muy difícil disponer de laboratorios con la capacidad necesaria para satisfacer las necesidades de la intensa práctica que la asignatura requiere.

En cuanto a las dificultades señaladas por los estudiantes, vinculadas con la relación con su par, amerita una profunda reflexión por parte del equipo docente y el desafío de encontrar mecanismos para evitar estas situaciones.

Desde lo institucional, este aporte de información puede apoyar el desarrollo de acciones orientadas a brindar a los estudiantes espacios físicos con las comodidades necesarias para que los pares puedan reunirse para estudiar, debatir y realizar las actividades prácticas.

En el marco de la asignatura, se realizará a futuro un análisis del desempeño de los equipos, comparando las calificaciones obtenidas por pares con las calificaciones obtenidas individualmente, e indagar sobre las dificultades de los alumnos que no pudieron superar la instancia de evaluación individual.

Referencias

[1] M. Guitert and F. Jiménez, "Aprender a colaborar," in Cooperar en clase: Ideas e instrumentos para trabajar en el aula, A. Campiglio and R. Rizzi, Eds. Madrid: M.C.E.P, 2000.

[2] R. M. Roque Hernández, R.González Morales, S. P.Muñoz Castellanos. "La programación por pares: un análisis de la producción científica en Web of Science".

[3] Forhum. International Journal of Social Sciences and Humanities, 4 (7), Julio-Diciembre 2022

[4] J. Dalton. "Great Big Agile. In Great Big Agile". Apress Berkeley, CA. 2019. https://doi.org/10.1007/978-1-4842-4206-3

[5] M. Maldonado, "El trabajo colaborativo en el aula universitaria," Laurus, vol. 13, no. 23, pp. 263–278, 2007.

Coordinadores:
- Cecilia Sanz (UNLP)
- Beatríz Depetris (UNTDF)
- Marcelo De Vincenzi (UAI)

CACiC 2023

CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

XII Workshop de Innovación en Educación en Informática

> Short Papers

WIEI

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Propuesta para la Enseñanza de Desarrollo de Software Guiado por Pruebas

Nicolás Paez[1], Alejandra Zangara[2], Diego Fontevila[1], and Alejandro Oliveros[1]

[1] Departamento de Ciencia y Tecnología, Universidad Nacional de Tres de Febrero, Sáenz Peña, Argentina

[2] Facultad de Informática, Universidad Nacional de La Plata, La Plata, Argentina

{nicopaez@computer.org, azangara@info.unlp.edu.ar, dfontdevila,aoliveros}@untref.du.ar

Abstract. El Desarrollo Guiado por Pruebas es una técnica de desarrollo ágil de software muy conocida pero de poco uso en la industria. Algunos autores sugieren que esto puede ser consecuencia de una falta de capacitación o tal vez de una enseñanza muy superficial y poco efectiva. A partir de esta problemática el presente artículo presenta los avances realizados de una tesis de maestría en Tecnología Informática Aplicada en Educación cuyo objetivo es generar una propuesta para la enseñanza del Desarrollo de Software Guiado por Pruebas. Este artículo presenta la motivación del trabajo de tesis, una breve descripción de sus aportes, el grado de avance logrado hasta el momento y los trabajos futuros de cara a completar la tesis.

Keywords: Desarrollo Guiado por Pruebas, Ingeniería de Software, TDD, Educación

1. Motivación

El Desarrollo Guiado por Pruebas, habitualmente conocido como TDD (del inglés Test-Driven Development) es una técnica de desarrollo de software creada por Kent Beck durante los años 90 y formalizada en su libro Test-Driven Development by Example de 2003 [1].

Luego de la publicación del mencionado libro la técnica fue ganando cada vez más popularidad en términos de cantidad de gente que la conoce y dando origen a diversas variantes, algunas de ellas con beneficios adicionales pero todas ellas sin mayor grado de adopción. Entre estas variantes cabe destacar BDD (del inglés Behaviour-Driven Development) [2] y SBE (del inglés Specification by Example)

[3].

TDD ha logrado una gran popularidad en términos de conocimiento tanto a nivel industria como academia. Se enseña en las universidades [4] y se han realizado diversas investigaciones formales sobre sus beneficios y propiedades [5–7]. Pero curiosamente dicha popularidad no se ha traducido en adopción, sino que es en cierto modo ”solo fama”. Dicho de otro modo: mucha gente conoce la técnica, sabe de su existencia, pero son muy pocos los que la utilizan en el desarrollo de software a nivel industrial[8] . Los estudios más optimistas reportan que TDD es utilizado por menos del 30 % de los programadores. Pueden pensarse diversas explicaciones para este fenómeno, algunas de ellas podrían ser: que es una técnica de muy difícil aplicación o que resulta muy

difícil aprenderla. He aquí la motivación del trabajo de investigación descripto en el presente artículo.

Dicho trabajo pertenece a una tesis para la Maestría en Tecnología Informática Aplicada en Educación de la Facultad de Informática de la Universidad Nacional de La Plata. El mismo es llevado a cabo por el tesista Ingeniero Nicolás Paez bajo la dirección de la Dra. Alejandra Zangara, la co-dirección del Dr. Diego Fontdevila y el asesoramiento técnico del Lic. Alejandro Oliveros. El trabajo de investigación fue iniciado en el año 2021 y se espera sea completado en el transcurso de 2023.

El objetivo del trabajo es desarrollar una Propuesta para la Enseñanza del Desarrollo Guiado por Pruebas.

A continuación, se describen los puntos principales del plan de trabajo:

1. Realizar una revisión formal de bibliografía para establecer el estado del arte en lo referente a la enseñanza de TDD
2. Relevar con especialistas en la materia desafíos y recomendaciones para la enseñanza de TDD
3. Elaborar una propuesta de enseñanza de TDD
4. Implementar la propuesta en al menos dos cursos para validarla

En la siguiente sección se detalla el estado actual de cada uno de los items del plan de trabajo.

2. Aporte del trabajo

En primera instancia, acorde a lo planificado, se realizó una revisión sistemática de literatura que será publicada como un artículo independiente en el transcurso del corriente año. Esta revisión sistemática representa un primer aporte del trabajo de investigación: conocer el estado del arte en la enseñanza de TDD.

En segunda instancia se llevó a cabo un relevamiento de desafíos y recomendaciones para la enseñanza de TDD. Este segundo relevamiento se hizo a partir de entrevistas individuales semi-estructuradas con 12 expertos en la enseñanza de TDD. Estas entrevistas, o más precisamente las información extraída y sintetizada de las mismas representa un segundo aporte de la tesis: un conjunto de desafíos y recomendaciones para la enseñanza de TDD. Este aporte también está planificado como una publicación independiente.

A partir de conocer el estado del arte, los desafíos y recomendaciones de los expertos se elaboró una propuesta para la enseñanza de TDD. En la elaboración de dicha propuesta también se ha tenido en cuenta la experiencia personal del tesista en la enseñanza de TDD. La parte conceptual de la propuesta de enseñanza ya se encuentra desarrollada y se está trabajando en la generación de los materiales didácticos que permitirán su implementación. Al mismo tiempo parte de la propuesta ya se está aplicando en las materias que actualmente dicta el tesista.

En términos resumidos, la propuesta para la enseñanza de TDD está estructurada en los siguientes pilares referentes al recorrido del contenido:

- Comenzar la enseñanza de TDD como un enfoque de programación resolviendo problemas de modelado y algoritmia utilizando el estilo ”Clásico de TDD” (también

denominado estilo Chicago) [9]

- Luego en segunda instancia, trabajar sobre problemas más complejos, incluyendo la problemática de manejo de entrada/salida y utilizando el estilo ”mockist” (conocido también como estilo London) [10]
- Adicionalmente, en esta segunda instancia, introducir otras prácticas de la metodología Extreme Programming (XP) que es la que dio origen a TDD. Concretamente las prácticas a incorporar son Programación de a Pares (Pair Programming) e Integración Continua (Continuous Integration) que son dos prácticas íntimamente relacionadas a TDD [11]

Esta parte de la propuesta ya ha sido presentada en un artículo que fue aceptado para publicación en el Simposio Argentino de Educación en Informática* que se llevará a cabo en el mes de Septiembre de 2023.

Por otro lado, en lo referente a la estrategia didáctica la propuesta plantea los siguientes pilares:

- Aula invertida: esta técnica propone sacar de la clase sincrónica el contenido teórico entregándolo a los alumnos en forma de videos y/o lecturas para que sea estudiado fuera del aula. Al mismo tiempo esto permite utilizar el tiempo de clase sincrónica para realizar actividades de aplicación utilizando estrategias de educación activa como las planteadas por Bowman [12]
- Aula extendida: esta técnica propone extender el espacio de intercambio más allá del aula de clase sincrónica a partir del uso de un aula virtual y demás herramientas digitales permitiendo un intercambio más fluido y continuo entre docentes y alumnos.
- Evaluación continua: este enfoque apunta a realizar evaluaciones de forma regular a lo largo de toda la duración del curso. Como estas evaluaciones son periódicas las mismas suelen ser más acotadas que las tradicionales evaluaciones parciales y al mismo tiempo suelen estar más enfocada en temas puntuales.
- Aprendizaje basado en proyectos: esta estrategia propone que los alumnos adquieran los conocimientos y competencias mediante la realización de proyectos que los enfrenten con situaciones del mundo real [13].

El uso de las técnicas enumeradas en la enseñanza de temas de Ingeniería de Software no es una novedad. Lo que sí resulta innovador en un punto es su uso combinado en un mismo curso. Parte de la propuesta para integrar estas estrategias ya ha sido descripta en una publicación anterior del tesista [14] y en términos resumidos consiste en dividir el dictado de la materia en dos partes de la siguiente forma:

- Una primera parte centrada en que los alumnos individualmente adquieran ciertos conocimientos y habilidades que les permitan desempeñarse de forma armónica y eficiente al trabajar en grupo. Esta primera parte se dicta en una modalidad de aula invertida y extendida y realizando evaluaciones continuas.
- Una segunda parte donde los alumnos trabajan en grupo en el desarrollo de un proyecto que les permita poner en práctica los conocimientos y técnicas estudiados en la primera parte de la materia. Todos los grupos deben presentar avances semanales que posibilitan una evaluación continua del desempeño de los grupos y

* https://52jaiio.sadio.org.ar/simposios/SAEI

sus integrantes.

3. Trabajos futuros

Una vez completos los materiales didácticos para la implementación de la propuesta la misma será implementada dos cursos de Ingeniería de Software de cara a validar su utilidad y efectividad. Esos cursos son: la asignatura Ingeniería de Software perteneciente a la carrera de Ingeniería de Computación de la UNTreF y la asignatura Métodos y Modelos de Ingeniería 2 perteneciente a la carrera de Licenciatura en Análisis de Sistemas de la UBA. Estas dos implementaciones de la propuesta serán utilizadas para su validación. Respecto del mecanismo de evaluación se utilizarán dos estrategias. En primer lugar, las distintas actividades de evaluación del curso permitirán verificar el grado de conocimiento de los temas logrado por los alumnos. Pero solamente esto podría no resulta suficiente pues podría haber alumnos que ya contarán con los conocimientos desde antes de tomar el curso (esto no sería raro ya que cierto porcentaje de los alumnos llegan al curso con experiencia en el ámbito laboral donde bien podrían haber aprendido los temas en cuestión). Es por esto que adicionalmente se realizará un cuasi-experimento. Se les pedirá a los alumnos resolver dos ejercicios: uno ejercicio al comienzo del curso y otro al final del mismo. Si el alumno no tenia conocimientos sobre los temas de la materia y los adquirió durante la cursada, entonces deberían apreciarse diferencias significativas en las soluciones del primer y segundo ejercicio.

Una completado el trabajo de tesis, podrían surgir algunos otros temas de investigación como ser la factibilidad de utilizar la estrategia didáctica planteada para enseñanza de otros temas de Ingeniería de Software.

References

1. Beck, K.: Test-Driven Development by Example. Addison-Wesley (2003)
2. Rose, S.,Wynne,M.,Hellesoy,A.The Cucumber for Java Book: Behaviour-Driven Development for Testers and Developers.United States:Pragmatic Bookshelf (2015)
3. Adzic, G.: Specification by Example: How Successful Teams Deliver the Right Software. Manning Publications (2011)
4. Paez, N., Oliveros, A., Fontdevila, D. Initial Assessment of Agile Development in the Undergraduate Curricula. In: Meirelles, P., Nelson, M., Rocha, C. (eds) Agile Methods. WBMA 2019. Communications in Computer and Information Science, vol 1106. Springer, Cham. https://doi.org/10.1007/978-3-030-36701-5_6
5. Buchan, J., Li L. and MacDonell, S. G., Causal Factors, Benefits and Challengesof Test-Driven Development: Practitioner Perceptions, 2011 18th Asia-Pacific Software Engineering Conference, Ho Chi Minh City, Vietnam, 2011, pp. 405-413, doi: 10.1109/APSEC.2011.44.
6. Crispin, L.. Driving Software Quality: How Test-Driven Development Impacts Software Quality, in IEEE Software, vol. 23, no. 6, pp. 70-71, Nov.-Dec. 2006, doi: 10.1109/MS.2006.157.
7. A. Causevic, D. Sundmark and S. Punnekkat, “Test case quality in test driven development: A study design and a pilot experiment,” 16th International Conference on Evaluation &

Assessment in Software Engineering (EASE 2012), Ciudad Real, 2012, pp. 223-227, doi: 10.1049/ic.2012.0029.

8. Paez, N., Fontdevila, D., Gainey, F., Oliveros, A. (2018). Technical and Organizational Agile Practices: A Latin-American Survey. In: Garbajosa, J., Wang, X., Aguiar, A. (eds) Agile Processes in Software Engineering and Extreme Programming. XP 2018. Lecture Notes in Business Information Processing, vol 314. Springer,
9. Cham. https://doi.org/10.1007/978-3-319-91602-6_10
10. Martin, B. Comparative Case Study: London vs. Chicago, https://bit.ly/ 3W1GtRl. Last accessed 8 Ago 2023
11. Freeman, S. Pryce, N.: Growing Object-Oriented Software, Guided by Tests. Pearson Education (1999)
12. Beck,K., Andres,C.. Extreme Programming Explained: Embrace Change.UnitedKingdom:Pearson Education (2004)
13. Bowman, S. L.. Training from the back of the room!:65 ways to Step aside and letthem learn. John Wiley & Sons, (2008)
14. Pérez, B. and Rubio, A.. A Project-Based Learning Approach for Enhancing´ Learning Skills and Motivation in Software Engineering. In Proceedings of the 51st ACM Technical Symposium on Computer Science Education (SIGCSE '20). Association for Computing Machinery, New York, NY, USA, 309–315. https://doi.org/10.1145/3328778.3366891
15. Paez, N. and De la Fuente, H., Software Engineering Education Meets DevOps: an Experience Report, 2022 IEEE Biennial Congress of Argentina (ARGENCON), San Juan, Argentina, 2022, pp. 1-7, doi: 10.1109/ARGENCON55245.2022.9940102.

Coordinadores:
- Patricia Pesado (UNLP)
- Elsa Estevez (UNS)
- Alejandra Cechich (UNComa)

CACiC 2023

CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

XX Workshop de Ingeniería de Software

> Full Papers

WIS

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Conflict resolution for aspect-oriented static modeling

Fernando Pinciroli[1], Laura Zeligueta[2] and Marcelo Palma[2]

[1] Instituto de Investigaciones, Universidad Nacional de San Juan, Argentina
fpinciroli@iinfo.unsj.edu.ar

[2] Research Institute, Universidad Champagnat, Mendoza, Argentina
{zeliguetalaura, mpalma}@uch.edu.ar

Abstract. Separation of concerns allows the achievement of important benefits in all phases of the software development life cycle. Thus, it is possible to take advantage of this technique with the consequent improvement of the understanding of the models. However, conflicts of different types may be produced when concerns are composed to form a single system, due to the same fact of having them managed separately. Additionally, this problem is increased by the number of people needed to deal with large projects. This article is focused on the composition of concerns in structural models, in which each concern is realized by an individual class diagram. In this paper, we present an experience in which four systems analysts elaborated six class diagrams which belong to a single system, and we expose the conflicts that occurred after the composition of the diagrams. After the analysis and classification of the conflicts, a set of modeling agreements and recommendations was elaborated in order to reduce them. Then, the models were rebuilt and composed again, with a significant decrease in the number of conflicts detected after the second composition.

Keywords: aspect-oriented models; class models; model composition; conflict resolution.

1. Introduction

The aspect-oriented paradigm offers a series of benefits that are described in a vast literature [1] [2] [3], and there is even a great amount of studies which show that, in the cases where this paradigm was used in real-world settings, those benefits were indeed achieved [4]. In addition, this approach solves the problems of scattering and tangling in the source code developed with object-oriented programming [5].

All the advantages are achieved due to modularization, a very important principle of software engineering [6] [7] where the aspect-oriented approach makes the most of keeping the crosscutting concerns separate throughout the entire software development life cycle [7]. Separation of concerns allows the achievement of these benefits not only in the programming phase, but also in all phases of the life cycle [8] [9]. Thus, it is possible to take advantage of the benefits of the reduction of size in software models, with the consequent improvement of the understanding of the models [10].

It might be the case that separation of concerns may generate conflicts of different types when concerns are composed to form a single system, as a result of having managed the concerns separately [11] [12] [13]. And this risk is increased by the number of people needed to deal with large projects.

In this article, we study the conflicts that arise after the composition of concerns in structural models. These models are generally made with class diagrams, and each

concern is realized by an individual class diagram [14]. Every concern usually contains classes that are also present in other diagrams, although with different members and relationships, due to the specific nature of the concern they realize. In a system of medium to high complexity, the intervention of several analysts is required. However, due to the different working styles they can have, many other conflicts may take place when all diagrams are composed into only one. We present an experience in which four systems analysts elaborated six class diagrams of a same system and we present the conflicts which occurred after the composition of the diagrams. Research in the field of software engineering requires the empirical study of phenomena that take place in the real world and, for this work, we have used a qualitative method called post-mortem analysis [15].

In Section 2, we briefly describe the process we have followed to develop the class models individually. Section 3 presents the strategy we chose to compose the models for the first time. In Section 4 we discuss the findings, focusing on the main conflicts detected after the first composition. Section 5 offers the modeling agreements which emerged after the analysis and classification of conflict types detected and the analysts used to rebuild their models. In Section 6 we present the new results after the second composition and, in Section 7, we provide the final results by comparing the conflicts between both compositions, with and without the modeling agreements. Finally, Section 8 presents our conclusions and the final thoughts with regard to future work.

2. Elaboration of class models

In order to conduct this study in the software development life cycle stage corresponding to static modeling, four systems analysts were summoned, each of which developed four class diagrams corresponding to three processes that belong to a biochemical laboratory. These three processes were selected from among the most complex of the 85 processes automated by the software system under study: manage protocol, transfer sample and receive sample.

Three analysts drew up one diagram each, while the fourth analyst, who we shall refer to as “expert analyst”, given his vast experience in the problem domain, drew up diagrams of the three processes. This distribution allowed us to compare the difference in the number of conflicts that occur if the diagrams are made by a single person or by a team. It was also ensured that each diagram has been developed by two analysts, to observe the conflicts that arise when models are built from two different points of view.

Each class diagram performs a single functional software requirement, in order to keep concerns separate throughout the entire life cycle, as proposed by various authors [14] [16] [17]. In the symmetric aspect-oriented approach, each model is equally important, as, as opposed to the asymmetric approach, in which a base portion of the system is considered to be affected by crosscutting concerns [18] [19].

After individual modeling, we obtained six class diagrams, and the number of classes shown in Table 1. The differences in the number of classes already suggest the appearance of a significant number of conflicts.

Table 1. Classes obtained.

Analyst	Concern		
	Manage protocols	Transfer samples	Receive samples
#1	50		
#2		23	
#3			18
#4	38	32	29

The analyst #4 -the expert analyst- modeled 99 classes to build the three class diagrams used for the experience. The entire model of the laboratory is made up of 202 classes, of which the concerns selected in this work cover approximately 50% of the whole model.

3. First composition of the models

The purpose of this work is to study the conflicts that may occur when several analysts intervene in the production of different parts of the static modeling within the same system, at the abstraction level of software requirements. Thus, we have produced a single model by applying a composition process on the six class diagrams developed by the different analysts.

The composition techniques can be classified into two categories: specification-based and heuristic-based techniques [20]. In specification-based techniques, the analysts explicitly specify the relationships among the elements of the source models before being composed [21] [22]; in heuristic-based techniques, the relationships are guessed by a set of predefined heuristics.

There are four proposed heuristic-based composition techniques: merge [14] [17] [23], override [14] [17] [24], select [14] and union [20]. In our work, we applied merge, but we made a variation to the original merge technique. Instead of overlapping the members when they have the same names, we duplicated them if they have different data types, which in turn avoids the "structural superimposition problem" highlighted by Tian et al. [25]. As a result, with this new alternative of composition, we now have five forms of composition: merge by overlapping, merge by duplication (the new one), override, select and union.

Manual composition is complicated and susceptible to human error [24], but we wanted to conduct it that way in order to accurately analyze every conflict or matching, classify each conflict and be able to design standard solutions to prevent or, at least, reduce their appearance.

Although the composition was carried out manually and guided by the expert analyst, we used Enterprise Architect to facilitate the detection of inconsistencies and avoid errors.

The composition process was as follows:

- We first composed 3 class diagrams out of the ones created by the expert analyst who drew up all the models. Some conflicts occurred at this stage, but all of them

were due to human error which, once corrected, resulted in a single, perfectly composed model, without conflicts.

- Then, we added the three remaining diagrams, one by one, by using the technique called "element matching" [26], and looked for elements with matching signatures to be composed.
- Every time we composed an element or a member, we updated the statistics.

The criteria for class composition were:

- We took a class and looked for it in the model by signature: name and {isAbstract} [27]. If it already existed, we considered it a "matching class", if not, an "added class".
- In the case of an "added class", we analyzed if it already existed in the model with a different name or if it did not exist at all. In the first case, we considered it, then, as a "conflicting class"; otherwise, we considered the class arose simply due to a different analysis.

The criteria for attribute composition were:

- All the attributes of the "added classes" were considered as arising from a different analysis.
- The attributes of the matching classes were classified into three types:
  - "Matching attribute": those whose signatures completely match [27]: name, data type, visibility and properties.
  - "Conflicting attribute": those that match in name but not in type, or that do not match in name but represent the same concept as another existing attribute.
  - "Added attribute": those that simply come from a different analysis.

Once the composition was completed, we obtained the results we shall present in the next section. There, we also analyze the causes of the production of added and conflicting classes and attributes.

4. Conflicts detected after the first composition

According to what was mentioned in Section 3, the modeling of the expert analyst who developed all the models should not have conflicts, except for some human errors we had already detected when reviewing the resulting models and that could easily be corrected. According to Bussard et al. [28] this strategy could prevent inherent conflicts, which can be also classified as semantic conflicts [25].

However, the composition of the models developed by the rest of the analysts did present different types of conflicts, which we describe below. All the conflicts detected could be classified as semantic conflicts.

Conflict #1: the same concept modeled in different ways. This is one kind of "redundancy" [29]. The analysts modeled the same reality in different ways, so different classes appeared in the composed model to represent the same concept. Table 2 shows the number of different classes, "added classes", with respect to the one designed by the expert analyst, and how many of them correspond to a different modeling style. The "Percentage of total" column indicates the fraction of added classes with respect to the total of classes. The "Percentage on added classes" column indicates the fraction of

classes that appeared due to a different modeling with respect to the total of added classes.

Table 2. Conflicts due to "added classes".

Analyst	Total classes	Added classes	Percentage of total	Different modeling	Percentage on added classes
#1	50	41	82.00%	25	60.98%
#2	18	13	72.22%	12	92.31%
#3	23	14	60.87%	3	21.43%
Total	**91**	**68**	**74.73%**	**40**	**58.82%**

Conflict #2: The same class named differently. This conflict can also be classified as "redundancy" (Table 3). Since the composition is made by combining elements with the same names, another conflict occurs when analysts use different names to label the same concepts. We call them "conflicting classes". France et al. propose the use of "pre-merge directives" to solve this issue [24].

Table 3. Conflicts due to "conflicting classes".

Analyst	Different classes	Different naming	Percentage
#1	41	16	39.02%
#2	13	1	7.69%
#3	14	11	78.57%
Total	**68**	**28**	**41.18%**

Conflict #3: Different classes with the same name. The other side of the conflict mentioned in the previous point corresponds to the use of the same name to name different classes. We did not detect any such cases, categorized as "naming conflicts on aspects" [25] or as "inconsistency" [29].

Conflict #4: Attributes of different classes. The different classes from the analysts have attributes that could be representing the same concepts, but which cannot be composed because they belong to different classes. Table 4 shows the number of attributes that cannot be composed.

Table 4. Conflicts due to "added attributes" belonging to "added classes".

Analyst	Total attributes	Attributes of different classes	Percentage
#1	99	72	72.73%
#2	65	51	78.46%
#3	61	33	54.10%
Total	**225**	**156**	**69.33%**

Conflict #5: The same attribute modeled in different ways. Again, this is another kind of "redundancy" type. The analysts modeled the same reality in different ways, so different attributes appear in the composite model to represent the same concept. Table 5 shows the number of attributes other than those specified by the expert analyst, and how many of them correspond to a different modeling style.

Table 5. Conflicts due to "added attributes" belonging to "matching classes".

Analyst	Different attributes	Different modeling	Percentage
#1	22	10	45.45%
#2	14	6	42.86%
#3	28	5	17.86%
Total	**64**	**21**	**32.81%**

Conflict #6: The same attribute named differently. This is another conflict of "redundancy". Since the composition is made by combining elements with the same name, another conflict occurs when analysts use different names to label the same concepts (attributes). This type of conflict is called "naming conflicts on object-oriented components" by Tian et al. [25]. Table 6 shows the number of identical attributes belonging to identical classes, but to which different names were assigned.

Table 6. Conflicts due to "conflicting attributes".

Analyst	Different classes	Different naming	Percentage
#1	22	12	60.98%
#2	14	8	92.31%
#3	28	23	21.43%
Total	**64**	**43**	**67.19%**

Conflict #7: Different attributes with the same name. As with classes, it would be possible for analysts to name different attributes with the same name. We also did not find this kind of inconsistency present in our composed models.

Conflict #8: Different styles of relationships. Because of the number of conflicts between classes are very high at this point, we considered it did not make sense to analyze conflicts in relationships among classes, an analysis we hope to carry out once we are able to reduce conflicts between classes.

5. Modeling agreements and second composition

Most conflict resolution methods require formal specifications [29] [30] [31] [32], while Sardinha et al. offer a tool that does not require formality but only detects conflicts in specifications of high-level requirements and not in classes [31]. Based on the types of conflicts detected, we developed a set of modeling agreements to avoid them or, at least, to reduce them. This experience was originally performed in Spanish and, as a result, many of the solutions are specific to that language. Undoubtedly, it will be necessary to adapt the modeling agreements to every different language.

In general, conflicts can be classified into syntactic and semantic categories [20]. In Table 7 we present the recommendations we elaborated to solve the conflicts that were detected after the first composition.

Table 7. Modeling agreements.

Scope	Type	Rule	Explanation
General	Syntactic	UML standard	UML 2.5.1 standard must be strictly followed.
Class	Semantic	Class instead of attribute	All attributes that could be modeled as a class must be represented that way.
	Syntactic	Association class	Model the association-class as a single class and add the association-class constraint to it.
	Semantic	Many-to-many class	The team must agree on a suitable name; if there isn't one, name the many-to-many class with the two names of the connected classes, in alphabetical order.
	Semantic	Generalization or "Type" class	Use generalization when the subtypes require different members or relationships among them. Use "Type" class when subtypes are not different among them or if they are just a list.
	Syntactic	Use of prepositions	Avoid the use of prepositions in class names.
	Syntactic	Use of accent mark	Avoid the use of accent marks (for languages other than English).
	Semantic	Use of adjectives	Avoid adjectives if possible. Use them if strictly necessary.
	Semantic	Use of plural	Avoid the use of plural names. Use them only if strictly necessary.
	Syntactic	Standard characters	Use only standard characters in class names.
	Syntactic	CamelCase	Class names must follow the CamelCase pattern.
	Semantic	Record	Add the word "Record" as a suffix to name classes with historical elements as objects (it could be a prefix in languages other than English).
	Semantic	Classes with items	Add the word "Item" as a suffix to name classes with items (it could be a prefix in languages other than English).
Attribute	Semantic	Data types	Define a standard set of data types: int, bool, char, etc.
	Semantic	Date and time	Use separate attributes for each concept.
	Semantic	Logical erase	Use the name "isActive" to indicate logical deletion.
	Semantic	Id#	Avoid the use of Id# attributes.
	Semantic	Derived attributes	Avoid the use of derived attributes.
	Semantic	Use of prepositions	Avoid the use of prepositions in attribute names.
	Syntactic	Use of accent mark	Avoid the use of accent marks (for languages other than English).
	Semantic	Use of adjectives	Avoid adjectives if possible. Use them if strictly necessary.
	Semantic	Use of plural	Avoid the use of plural names. Use them only if strictly necessary.
	Syntactic	Standard characters	Use only standard characters in attribute names.
	Syntactic	camelCase	Attribute names must follow the camelCase pattern.
	Semantic	Boolean attributes	Use a verb as a prefix in boolean attributes.
	Semantic	Completeness	Check all the attributes have data type.
	Semantic	Properties	Can be used as usual.

Scope	Type	Rule	Explanation
	Semantic	Standard names	Use standard names for the most common attributes. i.e.: - "name": for the object name. - "number": for the object number. - "description": for the explanation of the meaning of the object. - "comments": for comments, observations, etc. - Suffixes "From" and "To" for ranges. - "value": for value, price, etc. - "date": for the object creation date. - "surname": for family name. "name": in singular, for names.
Relation-ship	Semantic	Generalization	Can be used as usual, but the generalization: - must follow Liskow's principle. - must be checked against cyclic and conflicting inheritance and conflicting to avoid the "object-oriented composition problem [25]. will be flattened before the composition and abstract classes will disappear.
	Syntactic	Generalization lines	Join the generalization lines when the subtypes belong to the abstraction.
	Semantic	Composition and aggregation	Can be used as recommended in UML.
	Semantic	"1..1" multiplicity	Check if it can be "1..*" with the question: "Do we need history?".
	Semantic	"1..*" multiplicity	Check if it can be "0..*" with the question: "Can we find some case where the multiplicity can be 0?".
	Syntactic	Unknown multiplicity	Leave blank when the multiplicity is unknown.
	Semantic	Navigation	Can be used as usual.
	Semantic	Use of prepositions	Avoid the use of prepositions in relationship names.
	Syntactic	Use of accent mark	Avoid the use of accent marks (for languages other than English).
	Semantic	Use of adjectives	Avoid adjectives if possible. Use them if strictly necessary.
	Semantic	Use of plural	Avoid the use of plural names. Use them only if strictly necessary.
	Syntactic	Standard characters	Use only standard characters in relationship names.
	Syntactic	CamelCase	Class names must follow the CamelCase pattern.
	Semantic	Unary relationship	Use role names at each end of the relationship.
	Semantic	Constraints	Can be used as usual.

After our first composition, these modeling agreements were shared and explained within the team. Then, the analysts corrected their previous six class diagrams by applying the model agreements. These diagrams were composed again, following the same strategy:

- The composition technique was merged by duplication.
- We first composed the 3 class diagrams of the expert analyst.
- Then, we added the three remaining diagrams, one by one.
- Every time we composed an element or a member, we updated the statistics.
- Finally, we compared the statistics of both compositions.

6. Conflicts detected after the second composition

In this section we present the results obtained for every type of conflict after the second composition.

Conflict #1: The same concept modeled in different ways. The analysts modeled the same reality in different ways. As a result, different classes appeared in the composed model to represent the same concept. Table 8 shows the number of different classes, namely "added classes", with respect to those designed by the expert analyst, and how many of them correspond to a different modeling style. The "Percentage on added classes" column indicates the rate of classes due to a different modeling with respect to the total of classes that were added.

Table 8. Conflicts due to "added classes".

Analyst	Total classes	Added classes	Percentage of total	Different modeling	Percentage on added classes
#1	50	31	59.62%	28	90.32%
#2	18	13	72.22%	12	92.31%
#3	23	5	21.74%	3	60.00%
Total	**91**	**49**	**44.55%**	**43**	**87.76%**

Conflict #2: The same class named differently. Since the composition is made by combining elements with the same names, another conflict occurs when analysts use different names to label the same concepts. We shall refer to them "conflicting classes" (Table 9).

Table 9. Conflicts due to "conflicting classes".

Analyst	Different classes	Different naming	Percentage
#1	31	3	9.68%
#2	13	1	7.69%
#3	5	2	40.00%
Total	**49**	**6**	**12.24%**

Conflict #3: Attributes of different classes. The different classes among the analysts have attributes that could be representing the same concepts, but they cannot be composed because they belong to different classes (Table 10).

Table 10. Conflicts due to "added attributes" belonging to "added classes".

Analyst	Total attributes	Attributes of different classes	Percentage

#1	140	72	51.43%
#2	67	55	82.09%
#3	61	10	16.39%
Total	**268**	**137**	**51.12%**

Conflict #4: The same attribute modeled in different ways. The analysts modeled the same reality in different ways, so different attributes appear in the composite model to represent the same thing. Table 11 shows the number of attributes other than those of the expert analyst, and how many of them correspond to a different modeling style.

Table 11. Conflicts due to "added attributes" belonging to "matching classes".

Analyst	Different attributes	Different modeling	Percentage
#1	33	29	87.88%
#2	5	3	60.00%
#3	11	8	72.73%
Total	**49**	**40**	**81.63%**

Conflict #5: The same attribute named differently. Since the composition is made by combining elements with the same names, another conflict occurs when analysts use different names to label the same concepts (attributes). Table 12 shows the number of identical attributes belonging to identical classes, but to which different names were assigned.

Table 12. Conflicts due to "conflicting attributes".

Analyst	Different classes	Different naming	Percentage
#1	22	4	12.12%
#2	14	2	40.00%
#3	28	3	27.27%
Total	**64**	**9**	**18.37%**

Conflict #6: Redundancy conflicts. We were able to confirm the need to perform an activity after composing, which is called "post-merge" by different authors [24] [27].

In our experiment, non-derived attributes in their respective source diagrams became derived attributes when they were combined by the merging process. For example, "age" and "date of birth" together make the former derive from the latter. Then, after the composition of the class diagrams, we see as necessary to carry out a new activity to correct the redundancy conflicts that may have arisen.

7. Final results

After the second composition, we collected the differences found with respect to the first composition. It is possible to observe a substantial improvement after the application of the modeling agreements in the production of the models elaborated independently by the different analysts.

Conflict #1: The same concept modeled in different ways. Table 13 compares Tables 2 and 8 and shows that the number of added classes was reduced by 58.62%. We also obtained a 32.97% improvement because these classes are produced by a difference in the analysts' modeling style, which went from 58.85% to 87.76%. We believe that it would not be possible to improve this number with rules, but analysts should have prior training to align their modeling styles.

Table 13. Improvement after the second composition in "added classes".

Composition	Added classes	Different modeling
1st composition	74.73%	58.85%
2nd composition	44.55%	87.76%
Improvement	**58.62%**	**32.97%**

Conflict #2: The same class named differently. Table 14 compares Tables 3 and 9 and shows a significant improvement in the reduction of conflicts due to the different naming criteria of the classes, which were unified by the rules of the modeling agreement.

Table 14. Improvement after the second composition in "conflicting classes".

Composition	Different naming
1st composition	41.18%
2nd composition	12.24%
Improvement	**70.26%**

Conflict #3: Attributes of different classes. Table 15 compares Tables 4 and 10 and presents the logical reduction due to the lower number of added classes, as commented in point 7.1.

Table 15. Improvement after the second composition in "added attributes" belonging to "added classes".

Composition	Attributes of different classes
1st composition	69.33%
2nd composition	51.12%
Improvement	**26.27%**

Conflict #4: The same attribute modeled in different ways. As previously mentioned, the decrease of conflicts due to differences in conventions means that these differences lie in the diverse modeling styles of the analysts involved in the project, as can be seen in Table 16, which compares Tables 5 and 11.

Table 16. Improvement after the second composition in "added attributes" belonging to "matching classes".

Composition	Different attribute modeling
1st composition	32.81%
2nd composition	81.63%
Improvement	**59.80%**

Conflict #5: The same attribute named differently. Table 17 shows, once again, a significant improvement in the reduction of conflicts due to the different naming criteria of the attributes, which were unified by the rules of the modeling agreement. It compares Tables 6 and 12.

Table 17. Improvement after the second composition in "conflicting attributes".

Composition	Different attribute naming
1st composition	67.19%
2nd composition	18.37%
Improvement	**72.66%**

Finally, the matching classes and attributes increased notably. Table 18 summarizes our findings on this issue. The percentages of the matching classes and attributes were calculated with respect to the total of classes and attributes respectively.

Table 18. Matching classes and attributes.

Composition	Matching classes	Matching attributes
1st composition	23 out of 128	5 out of 225
2nd composition	44 out of 110	82 out of 268
Improvement	**55.08%**	**92.74%**

8. Conclusions

We have mentioned that separation of concerns throughout the software development life cycle has the possibility that conflicts of different types may be generated when concerns are composed to form a single system, given the fact that the concerns have been managed separately.

Since each concern is realized by an individual class diagram, all of them have shared classes, although with different members and relationships, due to the specific nature of the concern they realize. Therefore, in Section 4 we have described the conflicts that arose when all the diagrams were composed into only one due to the different working styles performed by different analysts. Thus, throughout our experiment, we were able to demonstrate that the need to establish modeling criteria and rules is critical, in order to reduce the amount and types of conflicts that occur when different analysts intervene to produce, separately, the models that will end up being composed in only one.

This article is part of a larger research project [37] where a series of research is being caried out concerning a framework process that has been developing for several years [36] [37] [38], although there are still many lines open for future work.

Regarding this, we believe there is a lot to do about pre and post composition activities [28]. We are thinking about deepening the study on the impact of changes we do during the composition: addition, removal, modification and derivation of elements in the composed model [21]. The production of modeling patterns will also be very useful. Rules and conventions for naming elements are crucial since the names of the modeling elements are the basis of composition. The study of the different conflict types and their classification will also be a great contribution to design techniques that allow them to be addressed and solved more effectively. The composition of relationships is also an important area to explore [30]. Finally, the automatization of modeling agreements, and the use of online collaborative model editors, will be of great help to obtain an efficient model composition.

References

[1] FanJiang Y, Kuo J, Ma S, Huang W (2010) An Aspect-Oriented Approach for Mobile Embedded Software Modeling. In: Taniar D, Gervasi O, Murgante B, Pardede E, Apduhan BO (eds.) Computational Science and Its Applications, ICCSA 2010, vol. 6017, Berlin, Heidelberg, pp. 257–272. doi: 10.1007/978-3-642-12165-4_21.

[2] Jalali A (2015) Static Weaving in Aspect Oriented Business Process Management. Conceptual Modeling, Stockholm, pp. 548–557.

[3] Singh N, Singh Gill N (2012) Towards an Integrated AORE Process Model for Handling Crosscutting Concerns. IJCA, vol. 37, no. 3, pp. 18–24. doi: 10.5120/4587-6525.

[4] Pinciroli F, Barros Justo JL, Forradellas R (2020) Systematic mapping study: On the coverage of aspect-oriented methodologies for the early phases of the software development life cycle. Journal of King Saud University - Computer and Information Sciences. doi: 10.1016/j.jksuci.2020.10.029.

[5] Kiczales G et al. (1997) Aspect Oriented Programming. Proceedings of the European Conference on Object-Oriented Programming (ECOOP), vol. 1241, p. 25.

[6] Parnas DL (1972) On the criteria to be used in decomposing systems into modules. Commun. ACM, vol. 15, no. 12, pp. 1053–1058. doi: 10.1145/361598.361623.

[7] Dijkstra EW (1982) Selected writings on computing: a personal perspective. Springer-Verlag, New York.

[8] Ye S, He C (2013) A comparison of methods for identification of early aspects. Proceedings of 2013 3rd International Conference on Computer Science and Network Technology, Dalian, pp. 275–279. doi: 10.1109/ICCSNT.2013.6967112.

[9] Rashid A, Moreira A (2006) Domain Models Are NOT Aspect Free. In: Nierstrasz O, Whittle J, Harel D, Reggio G (eds.) Model Driven Engineering Languages and Systems, vol. 4199, Berlin, Heidelberg, pp. 155–169. doi: 10.1007/11880240_12.

[10] Rashid A et al. (2010) Aspect-Oriented Software Development in Practice: Tales from AOSD-Europe. Computer, vol. 43, no. 2, pp. 19–26. doi: 10.1109/MC.2010.30.

[11] Vanoli VL, Marcos CA (2007) Early Conflicts: Análisis y Resolución de Conflictos Tempranos. XIX Encuentro Chileno de Computación, Jornadas Chilenas de Computación, Iquique, p. 1-14.

[12] Sardinha A, Araújo J, Moreira A, Rashid A (2010) Conflict Management in Aspect-Oriented Requirements Engineering. Information Sciences and Technologies Bulletin of the ACM Slovakia, vol. 2, no. 1, pp. 56-59.

[13] Pryor JL, Marcos C (2003) Solving Conflicts in Aspect-Oriented Applications. Proceedings of the Fourth Argentine Symposium on Software Engineering (ASSE´2003), Jornadas Argentinas de Informática e Investigación Operativa, vol. 32, Buenos Aires, pp. 1-10.

[14] Pinciroli F (2020) Proceso marco orientado a aspectos en las etapas tempranas del ciclo de vida del desarrollo de software para una transición en la industria. PhD tesis, Universidad Nacional de San Juan, San Juan.
[15] Wohlin C, Höst M, Henningsson K (2006) Empirical Research Methods in Web and Software Engineering. In: Mendes E, Mosley N (eds.) Web Engineering, Springer-Verlag, Berlin/Heidelberg, pp. 409–430. doi: 10.1007/3-540-28218-1_13.
[16] Jacobson I, Ng PW (2005) Aspect-oriented software development with use cases. Upper Saddle River, Addison-Wesley.
[17] Clarke S, Baniassad E (2005) Aspect-oriented analysis and design: the theme approach. Upper Saddle River, Addison-Wesley.
[18] Bálik J, Vranić V (2012) Symmetric aspect-orientation: some practical consequences. Proceedings of the 2012 workshop on Next Generation Modularity Approaches for Requirements and Architecture, NEMARA '12, Potsdam, p. 7. doi: 10.1145/2162004.2162007.
[19] Reddy YR et al. (2006) Directives for Composing Aspect-Oriented Design Class Models. In: Rashid A, Aksit M (eds.) Transactions on Aspect-Oriented Software Development I, vol. 3880, Berlin, Heidelberg, , pp. 75–105 . doi: 10.1007/11687061_3.
[20] Farias de Oliveira KS (2012) Empirical Evaluation of Effort on Composing Design Models. Pontificia Universidade Católica do Rio de Janeiro, Rio de Janeiro.
[21] Kolovos D, Rose L, García-Domínguez A, Paige RF (2022) The Epsilon Book. https://www.eclipse.org/epsilon/doc/book/EpsilonBook.pdf. Accessed 7 March 2022.
[22] Whittle J, Jayaraman P, Elkhodary A, Moreira A, Araújo J (2009) MATA: A Unified Approach for Composing UML Aspect Models Based on Graph Transformation. In: Katz S, Ossher H, France R, Jézéquel JM (eds.) Transactions on Aspect-Oriented Software Development VI, vol. 5560, Berlin, Heidelberg, pp. 191–237. doi: 10.1007/978-3-642-03764-1_6.
[23] Object Management Group: OMG® Unified Modeling Language® version 2.5.1. OMG® Unified Modeling Language®. https://www.omg.org/spec/UML/2.5.1/PDF. Accessed 7 March 2022.
[24] France R, Fleurey F, Reddy R, Baudry B, Ghosh S (2007) Providing Support for Model Composition in Metamodels. 11th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2007), Annapolis, pp. 253–253. doi: 10.1109/EDOC.2007.55.
[25] Tian K, Cooper K, Zhang K, Yu H (2009) A Classification of Aspect Composition Problems. 2009 Third IEEE International Conference on Secure Software Integration and Reliability Improvement, Shanghai, pp. 101–109. doi: 10.1109/SSIRI.2009.33.
[26] Kienzle J, Al Abed W, Jacques K (2009) Aspect-oriented multi-view modeling. Proceedings of the 8th ACM international conference on Aspect-oriented software development, AOSD '09, Charlottesville, p. 87. doi: 10.1145/1509239.1509252.
[27] Reddy R, France R, Fleuery F, Baudry B (2005) Model Composition - A Signature based approach. Proceedings Aspect Oriented Modeling Workshop, MODELS/UML, Montego Bay.
[28] Bussard L, Carver L, Ernst E, Jung M, Robillard M, Speck A (2000) Safe Aspect Composition. European Conference on Object-Oriented programming, ECOOP 2000, Sophia, Antípolis and Cannes.
[29] Elasri H, Elabbassi E, Abderrahim S (2018) Semantic integration of UML class diagram with semantic validation on segments of mappings, arXiv:1801.04482 [cs.SE].
[30] Laney R, Barroca L, Jackson M, Nuseibeh B (2004) Composing requirements using problem frames. Proceedings of the 12th IEEE International Requirements Engineering Conference, Kyoto, pp. 113–122. doi: 10.1109/ICRE.2004.1335670.
[31] Mostefaoui F, Vachon J (2007) Design-Level Detection of Interactions in Aspect-UML Models Using Alloy. JOT, vol. 6, no. 7, p. 137. doi: 10.5381/jot.2007.6.7.a6.

[32] Weston N, Chitchyan R, Rashid A (2008) A Formal Approach to Semantic Composition of Aspect-Oriented Requirements. 6th IEEE Intl. Req. Eng. Conference, Barcelona, pp. 173–182. doi: 10.1109/RE.2008.42.

[33] Petersen K, Vakkalanka S, Kuzniarz L (2015) Guidelines for conducting systematic mapping studies in software engineering: An update. Information and Software Technology, vol. 64, pp. 1–18. doi: 10.1016/j.infsof.2015.03.007.

[34] Pinciroli F, Zeligueta L, Palma M, Cappello I, Motta E (2021) Desarrollo de incumbencias en el modelado de la vista estática. XXIII Workshop de Investigadores en Ciencias de la Computación, WICC 2021, Chilecito.

[35] Pinciroli F (2015) AOP4ST – Aspect-Oriented Process for a Smooth Transition. Proceedings of the XVII Workshop de Investigadores en Ciencias de la Computación, WICC 2015, Salta, p. 5.

[36] Pinciroli F, Barros Justo JL (2017) Early aspects in 'Aspect-Oriented Process for a Smooth Transition'. XIV Workshop Ingeniería de Software (WIS), Congreso Argentino de Ciencias de la Comp., CACIC 2017, La Plata.

[37] Pinciroli F (2019) Modeling the Static View in Aspect-Oriented Software Development. Proceedings of the III International Congress on Computer Sciences and Information Systems, CICCSI 2019, Mendoza.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Addressing the State Explosion Problem for Big Data Systems Formal Verification

Fernando Asteasuain[1,2]

Universidad Nacional de Avellaneda, Argentina
fasteasuain@undav.edu.ar
Universidad Abierta Interamericana - Centro de Altos Estudios CAETI, Argentina

Abstract. The formal verification of BIG DATA systems remains as a challenging task to be addressed since a very large and complex state space describing the behavior of the system must be explored and verified. In particular, the state explosion problem arises as one of the most problematic issues to be faced against. Some approaches have leveraged on some architectural patterns used in BIG DATA system development, especially those focused on the MAP-REDUCE architecture. Taking this into consideration in this work we present VG-FVS, a new version of our framework FVS (Feather weight Visual Scenarios), which is specially developed to address the state explosion problem. This is achieved by integrating FVS with MaRDiGraS, a generic library which eases the state space exploration using a MAP-REDUCE software architecture. Empirical validation analyzing BIG DATA systems was carried on, showing promising results for our approach.

Keywords: Formal verification, BIG DATA Systems, State Explosion

1. Introduction

Modern software development sometimes includes the outstanding presence of a vast amount of data to be analyzed, explored and inspected. Valuable information and analysis can be automatically gathered employing several artificial intelligence techniques. This phenomena may be the origin of the so called *Data Science Software Development* [12,19,17]. It is clear that systems in this domain also need to be properly formally verified. That is, their expected behavior must be first specified and then it must be verified that the implementation fulfills the specification. Common formal verification techniques and tools were adapted to fit the new challenges that these kind of systems impose [5,13,8,18,11,20,15].

In this sense, the most usual extensions in the software verification phase were focused on improving performance issues by parallelising tools and algorithms or enriching the expressive power of the specification language [5,8,2,16,9]. When advancing in this direction, and old “enemy” of formal software verification must be faced: *the state explosion problem*[21]. This problem can be stated in simple words in the following manner: the number of possible states delineating the behavior of the system under analysis is way too extensive to apply formal verification tools. Perhaps the MaRDiGraS framework [4] represents one of the most meaningful milestones to address this problem. This framework employs a very general formalism to simplify the construction of very large state transition systems on large clusters and cloud

computing platforms. It is based on the Hadoop MapReduce architecture [10,23], which makes it specially suited for Data Science Software Verification.

One of the most distinguished characteristics of this tool is that its mains software components can be easily extended to cope with different formalisms. For example, in [4] it is shown how several formalisms based on Petri Nets can be adapted and used in the MaRDiGraS tool. Taking this fact into consideration in this work we present a novel extension of our behavioral specification tool FVS (Feather Weight Visual Scenarios) [1–3]. **This extension, named VGFVS, serves the purpose of integrating FVS with MaRDiGraS**. FVS is a very simple yet powerful behavioral specification language based on visual scenarios depicting the expected behavior of the system under analysis. It has been extended to cope with temporal and branching behavior, as well as being able to synthesize behavior [3]. A parallel version of FVS integrated with a parallel model checker was also previously introduced in [2]. We now build on the top of these extensions introducing the *VG-FVS version*, integrating FVS with the MaRDiGraS tool so that now FVS is also being able to tackle the state explosion problem. Our approach is empirically validated by analyzing several examples found in the literature considering Data Science and Big Data Systems [5]. These results show a promising future for FVS to formally validate Data Science and BIG Data Systems.

The rest of this work is structured as follows. Section 2 briefly introduces FVS's main features whereas Section 3 describes FVS's formal characterization. This is needed in order to undertand how FVS is integrated with MaRDiGraS, a procedure which is detailed in Section 4. Section 5 exhibits the empirical validation of our approach. Finally, Section 6 presents future and related work and enumerates some final conclusions.

2. Feather weight Visual Scenarios

In this section we will informally describe the standing features of FVS. The reader is referred to [1] for a formal characterization of the language. FVS is a graphical language based on scenarios. Scenarios are partial order of events, consisting of points, which are labeled with a logic formula expressing the possible events occurring at that point, and arrows connecting them. An arrow between two points indicates precedence. For instance, in Figure 1-(a) *A*-event precedes *B*-event. In Figure 1-b the scenario captures the very next *B*-event following an *A*-event, and not any other *B*-event. Events labeling an arrow are interpreted as forbidden events between both points. In Figure 1-c *A*-event precedes *B*-event such that C-event does not occur between them. Finally, FVS features aliasing between points. Scenario in 1-d indicates that a point labeled with *A* is also labeled with $A \wedge B$. It is worth noticing that *A*-event is repeated on the labeling of the second point just because of FVS formal syntaxis.

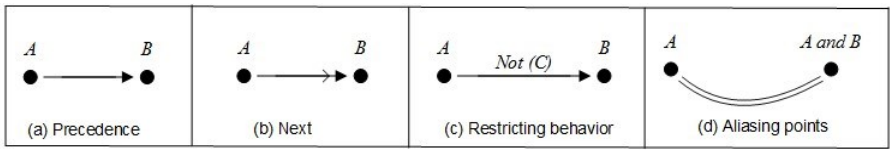


Fig.1. Basic Elements in FVS

We now introduce the concept of FVS rules, a core concept in the language. The intuition is that whenever a trace "matches" a given antecedent scenario, then it must also match at least one of the consequents. In other words, rules take the form of an implication: an antecedent scenario and one or more consequent scenarios. Graphically, the antecedent is shown in black, and consequents in grey. Since a rule can feature more than one consequent, elements which do not belong to the antecedent scenario are numbered to identify the consequent they belong to. An example is shown in Figure 2. The rule describes a very simple requirement from a communication protocol to be embedded on an onboard satellite computer: if some data is ready to be transmitted and the satellite is connected, then transmission can begin.

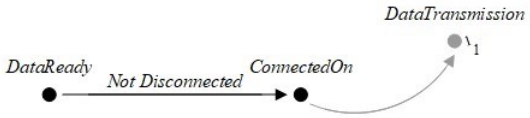


Fig.2. An FVS rule example

3. FVS Formal Definitions

In this section we sketched the most representative formal definitions of the FVS language. For a more comprehensive understanding of FVS's formal syntax and semantic please see [1]. These definitions are necessary to understand how FVS con be embedded into the MarDIGRaS formalism and tool. We need to define the following FVS concepts: *scenarios, morphisms, FVS translation into Büchi automata, how states are represented in FVS, and how state's successors are calculated*. With these definitions in mind the inclusion of FVS into MaRDiGraS can be straightforwardly delineated.

We first define the concept of scenarios.

Definition 1 (FVS Scenario). *An FVS* scenario *is a tuple* $(\Sigma, P, l, \equiv, \not\equiv, <, \gamma)$ *where:*

S1: Σ *is a finite set of propositional variables standing for types of events such that* $\Sigma = \Sigma_c \cup \Sigma_{uc}$ *where* Σ_c *represents controllable events and* Σ_{uc} *non controllable events*

S2: P *is a finite set of points;*

S3: $\ell: P \rightarrow \mathrm{P}\mathcal{L}(\Sigma)$ *is a function that labels each point with a given formula;*

S4: $\equiv \subseteq P \times P$ *is an equivalence relation;*

S5: $\not\equiv \subseteq P \times P$ *is an asymmetric relation among points;*

S6: $< \subseteq (P \uplus \{\mathbf{0}\} \times P \uplus \{\infty\}) \setminus \{(0, \infty)\}$ *is a precedence relation between points, where* $\mathbf{0}$ *and* ∞ *represent the beginning and the end of execution, respectively; S7:* $\gamma : (\not\equiv \cup <) \rightarrow P\mathcal{L}(\Sigma)$ *assigns to each pair of points, related by precedence or separation, a formula which constrains the set of events occurrences that may occur between the pair.*

We now formally define morphisms between scenarios. Intuitively, we would like to obtain a matching between scenarios ,i.e., a **mapping** between their points exhibiting how a scenario "specializes" another one [7].

Definition 2 (Morphism). *Given two scenarios* S_1, S_2 *(assuming a common universe of event propositions), and f a total function between* P_1 *and* P_2 *we say that f is a morphism from* S_1 *to* S_2 *(denoted* $f : S_1 \rightarrow S_2$*) iff*

M1: $\ell_2(a) \Rightarrow \ell_1(p)$ *is a tautology for all* $p \in P_1$ *and all* $a \in P_2$ *such that* $a \equiv_2 f(p)$;

M2: $\gamma_2(f(p), f(q)) \Rightarrow \gamma_1(p,q)$ *is a tautology for all* $p, q \in P_1$;

M3: *if* $p \equiv_1 q$ *then* $f(p) \equiv_2 f(q)$ *for all* $p, q \in P_1$;

M4: *if* $p \not\equiv_1 q$ *then* $f(p) \not\equiv_2 f(q)$ *for all* $p, q \in P_1$;

M5: *if* $p <_1 q$ *then* $f(p) <_2 f(q)$ *for all* $p, q \in P_1$.

3.1 Tableau Algorithm: Translating FVS Scenarios into Büchi Automata

We now present some basic concepts to understand the tableau algorithm while the reader is referred to [1] for a more detailed version of it. From a formal point of view, FVS scenarios can be defined as morphisms from the antecedent to the consequent. The algorithm relies on the notion of situations [1]. In few words, a situation represents for a given rule possible combinations of partial matches from the antecedent to the consequent. Consider the following example in figure 3. In this case, a rule with two consequents is shown. Furthermore, there are three partial matches for consequent one, and two for consequent two. Therefore, η_1 consists of the three morphisms in the first column (g_1^1, g_1^2, g_1^3), whereas η_2 consists of the two morphisms in the second column (g_2^1, g_2^2).

Given a rule *R*, the tableau builds a Büchi Automaton B = $\langle \Sigma, S, S^0, \Delta, F \rangle$ such that Σ constitutes minterms over Σ_R and the set of states S are triples ($Y_R \times bool \times P\mathcal{L}(\Sigma_R)$), where $P\mathcal{L}(\Sigma)$ is a function that labels each point with a given formula. The set Y_R associated to a state (a set of situations η), denoted *situations*(S), symbolically represents all the possible combination of partial matches obtained up to that state from the antecedent to each consequent. The second term of the triple identify accepting states. This boolean variable is set to **true** when the pattern is completely matched and will make the state transient. Finally, a third element is needed to maintain future obligations of the trace. These formulas are needed when rules predicate about conditions that must hold until the end of the trace.

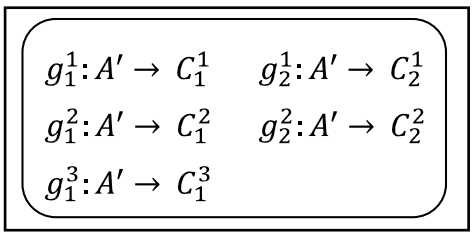


Fig.3. Asituationexample

The pseudo-code sketched in Algorithm 1 computes the successor states for transition relation Δ. Starting from the initial state (⟨∅,false,true⟩), the automata will try to incrementally "construct" the pattern as events, represented by minterms, occurs. For

every minterm, algorithm 1 computes all possible matchings considering matchings in the antecedent and also in each consequent. This is obtained trough two auxiliary algorithms, *advanceAntecedent* (line 5) and *advanceConsequent* (line 6). Line 7 analyzes if any successor reaches a *trap situation*, a situation where the antecedent has been matched (a morphism such that $A' = A$), but matching for all consequents is known unfeasible. A configuration of an scenario stands for a valid set of events occurring in the scenario.Lines 8 and 9 check if any consequent has been matched by the last move. This is, *goalmatched*[i] = *true* if and only if *consequent* C_i is matched. Line 10 analyzes if the next state is an accepting state: a consequent has been matched and it is not a trap situation. Finally, line 11 returns the expected output.

4. VG-FVS: FVS meets MaRDiGraS

In this section we show how FVS is connected to the MaRDiGraS tool. In order to achieve this, we first explain MaRDiGraS main concepts (Section 4.1) and latter on, how them are extended to cope with FVS definitions (Section 4.2).

4.1 MaRDiGraS

As explained in [4] MaRDiGraS is a distributed software tool aimed to built big states spaces for different kinds of formalisms. It is a very helpful tool since it is designed for simplifying the task of dealing with a large amount of reachable states by exploiting large clusters of machines. It was conceived as a generic library built on top of Hadoop MapReduce [10,23]. One of the most remarkable features of this tool is that its mains software components can be easily extended to cope with different formalisms. For example, in [4] it is shown how several kinds of Petri Nets can be adapted and used in the MaRDiGraS tool.

```
1   Algorithm Succ(S : State,m : minterm) : set of states;
2   Precondition : m ∧ obligations(S) is satisfiable;
3   newSits := ∅;
4   foreach η ∈ Situations(S) do
5       newSits := add(newSits,advanceAntecedent(η,m));
6       newSits := add(newSits,advanceConsequent(η,m));
7   trapSituation : ∃η ∈ newSits ∀i ∀j ∈ [1..n] g_j^i : A' → C_j^i ∈ η ∧ A' = A ∧ C_j^i it
    is not a configuration of C_j;
8   foreach j ∈ [1..n] do
        goalmatched[j] := ∃η ∈ situations(S) ∧
        g_j^i : A' → C_j^i ∈ η ∧ A' ⟶m C_j ∪ F_j^i ∧ m ∈ (R_F(C_j^i) ∧ C_j^i ∪ F_j^i = ∪ C_i;
9   goalMatched := (∃j (goalmatched[j])) ∧ (¬trapSituation) ;
10  return { ⟨newSits, GM, Obligations⟩ such that
11  GM → goalMatched ∧ GM = true → ∃j(goalmatched[j]) ∧ Obligations =
    Obligations(S) ∧ ⋀_{j∈I} R(C_j) ∧ GM = false → Obligations = Obligations(S)
```

Algorithm 1: Successor states

In the tool the behavior is given in terms of a labeled state transition system. The state explosion problem is addressed by building an abstraction of the original (concrete) state transition system. It is key in this process to properly define the notion of state, reachable states and successors states. Computation starts by considering the initial state of the system under analysis and goes on with a sequential state-space building phase until the set of states not yet explored becomes large enough, where "large enough" is a threshold to be defined for each system [4].

Architecturally, the MaRDiGraS contains two main artifacts: The Data component and the Core Component. The Data component consists of the "business" entities, such as the *State* or the *Edge* classes. This is the component to be addressed when trying to incorporate a given formalism into the tool. On the other hand, the Core components contains the implementation of all the algorithms that implements MaRDiGraS's main functioning.

In concrete, in order to adapt any formalism to this tool the following elements from the MaRDiGraS software must be extended:

- State: is an abstract class which should be extended to instantiate the state concept in a particular formalism.
- Successors: this method must return a list of new State objects representing the states directly reachable from the subject of the call. It must be defined for each formalism to be employed.
- IdentifyRelationship: this method must evaluate the actual relationship between (abstract) states sharing some specific features. The possible output values are: NONE, EQUALS, INCLUDED and INCLUDES.
- Edge: an abstract class which should be extended to represent the edge concept.
- GetFeatures: this functions analyses the equivalence between states.

4.2 VG-FVS: Adding FVS into the MaRDiGraS oily machinery

Given the formal definitions characterizing our FVS language given in Section 3 the combination of FVS with MaRDiGras is described as follows. We defined the MaRDiGrass **state class** as the states of the Büchi Automaton representing an FVS Scenario (see section 3.1). Similarly, the **Successors method** implementation follows the successors algorithm depicted in Algorithm 1. For the **Edge** class we implemented the notion of FVS morphisms. For the **IdentifyRelationship** method we implemented the notion of FVS configurations viewed as a set of FVS *situations*. This method returns NONE if the two configurations given as input do not match, returns EQUALS if they contain the same elements, INCLUDED in the case where the first configurations includes the second one but they are not the same one, and INCLUDES where the second configuration includes the first one but they are not the same one. Perhaps the most challenging implementation arose when dealing with the **GetFeatures** function. In this case, we implemented this function such as it returns the set of morphisms in each situation that fulfill the given consequent of the FVS rule. It was the only definition that requires some extra work since this concept was not included in the FVS formal characterization. Table 1 resumes the integration between FVS and MaRDiGraS.

Table 1. FVS-MaRDiGraS Integration

MaRDiGraS Element	FVS Element
State	FVS Büchi State
Successors	FVS Succ Method
Edge	Morphisms
IdentifyRelationship	Configurations Equivalence
GetFeatures	Morphisms Fulfilment

5. Experimentation Results

The empirical validation of the novel integration introduced in this work was based on previous experiments shown in [2], where we compared the parallel version of FVS combined with a distributed model checker against the technique presented in [5], which also addresses formal verification in big data systems. Also, we conducted the same case of studies and we compared not only execution times but also the space dimension. **In this occasion, we build on the top of that experiment comparing also the performance of *VG-FVS*.** We ran our experiments in a Bangho Inspiron5458, with a Dual Core i5-5200U and 8GB RAM memory. The analyzed case of studies, taken from [5], were a simple *Load Balancing System* and a *Shared Memory Example*, where clients try to gain access to a shared memory through a unique bus. Different configurations considering several number of clients were considered.

Table 2 shows the empirical comparison considering the execution time view. The column MAP-Reduce-CTL stands for the technique in [5], the column Parallel FVS stands for the previous version of our approach whereas the third column corresponds to VG-FVS. Execution times are expressed in seconds.

Table 2. Empirical Comparison: Execution Time

Example	Map-Reduce CTL	Parallel FVS	VG-FVS
5-Shared	110 sec	188 sec	165 sec
10-Shared	1032 sec	1341 sec	1005 sec
2-Load Balancing	43 sec	76 sec	71 sec
5-Load Balancing	114 sec	136 sec	120 sec
10-Load Balancing	18958 sec	20022 sec	18902 sec

Table 3 shows the empirical comparison considering the size dimension. In this case, we present the results for two concrete examples: the shared Memory example with 5 clients and the Load Balancing example with 5 clients. Similar results were obtained for the other versions of the examples considering more clients. For measuring size, we considered the number of states (*st* in Table 3) and transitions (*tr* in Table 3) of the automata involved in the verification process.

Table 3. Empirical Comparison: Size

Example	Map-Reduce CTL	Parallel FVS	VG-FVS
5-Shared	1863 st 7528 tr	3600 st 6500 tr	2160 st 3771 tr
5-Load Balancing	116175 st 425604 tr	301001 st 432653 tr	216720 st 302857 tr

Observations A few important notes can be taken from the empirical evaluation we conducted. Perhaps the most scintillating one is that using the new version of FVS the size of the automata involved was significatively reduced. In average, the size of the automata was reduced by a thirty percent, which represent a notable impact in the verification process. In addition, execution time was also reduced empowering the results of the validation of our approach. It must be noted that our results are still a bit worse than those ones produced by the technique [5]. However, this is somehow expected since our language is more expressive and flexible [1].

6. Conclusions and Future and Related Work

In this work we present VG-FVS, a novel extension of FVS to cope with the state explosion problem, a classic issue to be tacked when dealing with large systems. This is achieved by integrating FVS with the MaRDiGraS tool [4], a very well known software library specially built to address this problem. The results show the clear benefits of this integration, making FVS a very solid alternative to formally validate BIG DATA systems.

With respect to related work the most obvious comparison is against [5]. In this work the MaRDiGraS tool is also employed to formally validate distributed systems in the cloud. The empirical validation of our work shown in Section 5 is based on this approach ([5]). As a first difference we can say that our approach reflects nearly the same performance results while denoting a more expressive and flexible specification language. In the same line, we believe that the declarative nature of FVS specifications makes a more suitable approach to be adopted in the software industry. Work in [6] presents *Maude*, a technique to formally validate systems in the cloud. This tool is specially designated to validate cloud systems, while we pursue a more general objective. However, the architectural solution they provide it is a very skilful one. FVS could benefit introducing some of the architectural patterns employed in *Maude*. We would also like to interact with approaches like [22], which presents techniques in the BIG DATA Testing domain. In [14] an interesting approach based on micro services systems is presented. The technique explains how micro services architecture came in handy for big data systems' formal verification, a future line of work we would also like to address. In addition, we would like to provide a formal proof of the integration presented in this paper, assuring the correctness and soundness of this new FVS version.

References

1. F. Asteasuain and V. Braberman. Declaratively building behavior by means ofscenario clauses. *Requirements Engineering*, 22(2):239–274, 2017.

2. F. Asteasuain and L. R. Caldeira. A sound and correct formalism to specify, verify and synthesize behavior in big data systems. In *Communications in Computer and Information Science, vol 1584*, pages 109–123. http://doi.acm.org/10.1145/587051.587064, Springer, 2021.
3. F. Asteasuain, F. Calonge, M. Dubinsky, and P. Gamboa. Open and branchingbehavioral synthesis with scenario clauses. *CLEI E-JOURNAL*, 24(3), 2021.
4. C. Bellettini, M. Camilli, L. Capra, and M. Monga. Mardigras: Simplified buildingof reachability graphs on large clusters. In *RP workshop*, pages 83–95, 2013.
5. C. Bellettini, M. Camilli, L. Capra, and M. Monga. Distributed ctl model checkingusing mapreduce: theory and practice. *CCPE*, 28(11):3025–3041, 2016.
6. R. Bobba, J. Grov, I. Gupta, S. Liu, J. Meseguer, P. C. Olveczky, and S. Skeirik.¨ Survivability: design, formal modeling, and validation of cloud storage systems using maude. *Assured cloud computing*, pages 10–48, 2018.
7. V. Braberman, D. Garbervestky, N. Kicillof, D. Monteverde, and A. Olivero. Speeding up model checking of timed-models by combining scenario specialization and live component analysis. In *FORMATS*, pages 58–72. Springer, 2009.
8. M. Camilli. Formal verification problems in a big data world: towards a mightysynergy. In *ICSE*, pages 638–641, 2014.
9. A. Cerone and M. Roggenbach. *Formal Methods-Fun for Everybody*. Springer,
10. 2021.
11. J. Dean and S. Ghemawat. Mapreduce: simplified data processing on large clusters.*Communications of the ACM*, 51(1):107–113, 2008.
12. O. Hummel, H. Eichelberger, A. Giloj, D. Werle, and K. Schmid. A collection ofsoftware engineering challenges for big data system development. In *SEAA*, pages 362–369. IEEE, 2018.
13. M. Kim, T. Zimmermann, R. DeLine, and A. Begel. Data scientists in softwareteams: State of the art and challenges. *IEEE Transactions on Software Engineering*, 44(11):1024–1038, 2017.
14. V. D. Kumar and P. Alencar. Software engineering for big data projects: Domains,methodologies and gaps. In *IEEEBIGDATA*, pages 2886–2895. IEEE, 2016.
15. R. Laigner, M. Kalinowski, P. Diniz, L. Barros, C. Cassino, M. Lemos, D. Arruda,S. Lifschitz, and Y. Zhou. From a monolithic big data system to a microservices event-driven architecture. In *2020 SEAA*, pages 213–220. IEEE, 2020.
16. R. Laigner, M. Kalinowski, S. Lifschitz, R. S. Monteiro, and D. de Oliveira. A systematic mapping of software engineering approaches to develop big data systems. In *SEAA*, pages 446–453. IEEE, 2018.
17. C. Mandrioli, A. Leva, and M. Maggio. Dynamic models for the formal verificationof big data applications via stochastic model checking. In *2018 CCTA*, pages 1466– 1471. IEEE, 2018.
18. S. Mart´ınez-Ferna´ndez, J. Bogner, X. Franch, M. Oriol, J. Siebert, A. Trendowicz,A. M. Vollmer, and S. Wagner. Software engineering for ai-based systems: a survey. *TOSEM*, 31(2):1–59, 2022.
19. C. E. Otero and A. Peter. Research directions for engineering big data analyticssoftware. *IEEE Intelligent Systems*, 30(1):13–19, 2014.
20. A. Paleyes, R.-G. Urma, and N. D. Lawrence. Challenges in deploying machinelearning: a survey of case studies. *ACM Computing Surveys*, 55(6):1–29, 2022.
21. P. A. Sri and M. Anusha. Big data-survey. Indonesian Journal of Electrical Engineering and Informatics (IJEEI), 4(1):74–80, 2016.
22. A. Valmari. The state explosion problem. In *Advanced Course on Petri Nets*, pages 429–528. Springer, 1996.

23. A. Vianna, F. K. Kamei, K. Gama, C. Zimmerle, and J. A. Neto. A grey literaturereview on data stream processing applications testing. *Journal of Systems and Software*, page 111744, 2023.
24. J. Zhang and M. Lin. A comprehensive bibliometric analysis of apache hadoopfrom 2008 to 2020. *IJICC*, 16(1):99–120, 2023.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Gestión de los requerimientos no funcionales en equipos ágiles: un caso de estudio

Lourdes Romera, Jeremías González, Andrea Lezcano,
Juan A. Carruthers, Emanuel Irrazábal

Grupo de Investigación en Calidad de Software, FaCENA, UNNE, Argentina
{lromera,jegonzalez,alezcano,jacaruthers,eirrazabal}@exa.unne.edu.ar

Resumen. La gestión de los requerimientos no funcionales en un proyecto software desarrollado con enfoque ágil no siempre se hace de explícita. Esto disminuye la calidad de los resultados y una menor satisfacción del cliente. El trabajo propone la evaluación con estudio de caso de una guía para la gestión de requerimientos no funcionales en equipos ágiles de desarrollo software. La guía se enfoca en introducir dentro del flujo de trabajo artefactos y prácticas destinadas a apoyar y mejorar la gestión de este tipo de requerimientos. Como resultado de ello se observó una mejora en la identificación efectiva de los requerimientos no funcionales, aumentando la estabilidad del desarrollo y la comunicación con el cliente, sin disminuir la agilidad.

Palabras clave: Scrum, historias de sistema, requerimientos no funcionales.

1. Introducción

El cumplimiento de los plazos de entrega de software se ha vuelto un factor crítico para garantizar el éxito de los proyectos. En ese marco, el desarrollo ágil de software (ASD, por sus siglas en inglés) tiene como objetivo entregar continuamente software de valor, siendo el primero de los doce principios ágiles la satisfacción de las necesidades del cliente [1]. Uno de los pasos iniciales para ello es la redefinición de estas necesidades como requerimientos del proyecto, los cuales se dividen en dos categorías: requerimientos funcionales (RF) y requerimientos no funcionales (RNF). Los primeros describen las acciones específicas que el software debe realizar, y los segundos expresan las propiedades de calidad deseadas en un sistema. Estos aspectos aluden a qué tan bien funciona una aplicación o servicio y qué tan bien lo experimentan los usuarios (como, por ejemplo, su disponibilidad, facilidad de uso, confiabilidad o seguridad) [2]; también pueden ser restricciones de desarrollo, como el tiempo del proyecto, su costo, los recursos con los que deben llevarse adelante, los procesos que se deben utilizar o la tecnología de construcción [3].

En este sentido, es de vital importancia el tratamiento adecuado de los aspectos de calidad para la satisfacción de las expectativas del cliente. Según los resultados del mapeo sistemático proporcionado por Woubshet Behutiye [4] no existe un gran número de procedimientos, pautas o buenas prácticas para respaldar la documentación de los RNF y las pocas halladas tienden a abordar tipos específicos de RNF, como la usabilidad o la seguridad. A partir de los resultados obtenidos en el mapeo, se evidencia la necesidad de crear un conjunto de prácticas para la gestión de los RNF, teniendo en cuenta las perspectivas de los diversos actores y artefactos involucrados en ASD.

Estas prácticas deben estar basadas en evidencia empírica y evaluadas para respaldar eficazmente a los profesionales a la vez que no deben comprometer la agilidad del proceso de desarrollo de software, en términos de velocidad y simplicidad.

Este trabajo propone la evaluación mediante un estudio de caso de una primera aproximación a una guía para la gestión de RNF en equipos que tienen un enfoque ágil de desarrollo software, como paso previo a la construcción de un conjunto de procedimientos. La guía se enfoca en introducir dentro del flujo de trabajo de ASD artefactos y prácticas destinadas a apoyar y mejorar la gestión de los RNF.

El documento está organizado de la siguiente manera: en la sección 2, se presenta el estudio de caso llevado adelante en una pequeña empresa de desarrollo de software. En la sección 3 se exponen los resultados del estudio de caso y en la sección 4 se discuten las preguntas de investigación. Finalmente se concluye el trabajo en la sección 5.

2. Diseño del Estudio de Caso

Este trabajo propone evaluar el impacto de una guía para la gestión de requerimientos no funcionales en el flujo de trabajo de un equipo ágil en una pequeña empresa de desarrollo de software. Con este fin se formulan las siguientes preguntas de investigación (PI):

- PI1: ¿Hasta qué punto la guía facilitó la identificación temprana de los requerimientos no funcionales?
- PI2: ¿Se ha observado una mejora en la satisfacción del cliente al utilizar la guía en el proceso de gestión de requerimientos no funcionales?
- PI3: ¿La inclusión de nuevas prácticas y artefactos de trabajo afectó el grado de agilidad de los equipos de trabajo?

Se optó por emplear un enfoque de estudio de caso para responder las preguntas planteadas y abordar la investigación. Siguiendo la metodología propuesta por Yin [5], se llevó a cabo los siguientes pasos: diseño del estudio de caso, recopilación de datos, análisis de los datos recopilados y presentación de resultados. Cada uno de estos pasos se describen detalladamente en las siguientes subsecciones.

2.1 Diseño del Estudio

El estudio de caso se centra en una pequeña empresa de desarrollo de software de la región nordeste de Argentina con 12 empleados divididos en cuatro equipos. Dos de estos equipos adoptan un enfoque ágil de gestión de proyectos con Scrum, aunque actualmente no cuentan con experiencia previa en el manejo de requerimientos no funcionales. Este enfoque se ha seleccionado con el propósito de evaluar el impacto real de la guía y observar cómo su implementación afecta el proceso de obtención y gestión de requerimientos no funcionales. Para ello, el estudio se compone de cuatro fases descritas a continuación:

- **Fase 1:** Elaboración y uso de la guía para la gestión de requerimientos no funcionales en los equipos de trabajo.

- **Fase 2:** Análisis de la cantidad y calidad de los requerimientos no funcionales entre los proyectos previos y posteriores a la implementación de la guía.
- **Fase 3:** Análisis de los resultados entregados y el mantenimiento posterior.
- **Fase 4**: Medición del grado de agilidad de los equipos de desarrollo antes y después de aplicar la guía con la encuesta propuesta en [6].

2.2 Desarrollo de las Fases y Recopilación de los Datos

En esta sección, se desarrollaron una serie de actividades específicas para cada fase propuesta en la sección de diseño del estudio de caso. La empresa participante se involucró en estos pasos como parte integral del proceso de elaboración y evaluación del impacto de la guía. Cada acción se realizó en estrecha colaboración con el equipo de desarrollo y analistas no funcionales involucrados en los proyectos seleccionados, asegurando una recopilación coherente y completa de la información relevante para el estudio.

Fase 1. En la Fig. 1 se presenta el diagrama del ciclo de vida de un proyecto desarrollado con Scrum tal y como lo describe Van den Broek [7], incluyendo aspectos de gestión de pruebas. Este es el modelo seguido por los equipos de trabajo en la empresa de desarrollo objeto de estudio. Las leyendas en rojo indican los elementos incorporados en la guía, abarcando la gestión de los RNF para las etapas: iniciación del proyecto, Sprint inicial, planificación del Sprint y ejecución del Sprint. A continuación, a modo de resumen de la guía se presentan los elementos incorporados en cada etapa.

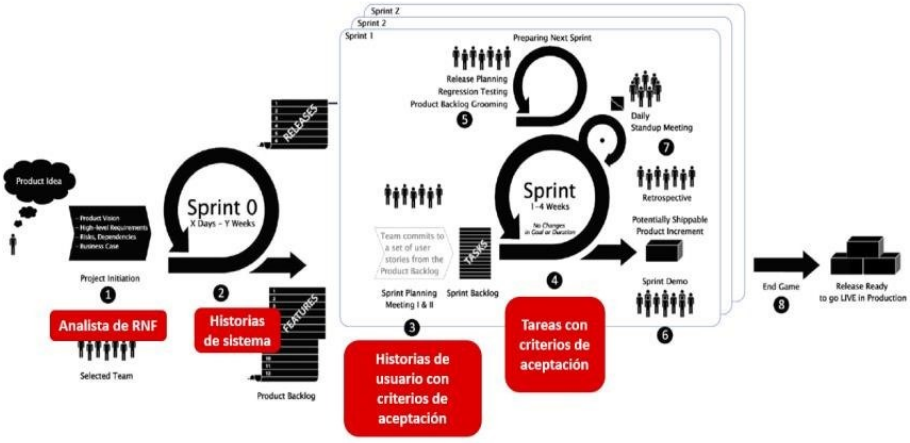


Fig. 1. Ciclo de vida de Scrum con las modificaciones.

Etapa de Inicio del Proyecto. Los roles tradicionales de Scrum no siempre poseen las habilidades necesarias para definir RNF específicos. La recomendación incluida en la guía es que un miembro del equipo se ocupe exclusivamente del análisis exhaustivo de los RNF, evaluando su impacto en el sistema, identificando las dependencias con los requerimientos funcionales y realizando pruebas para validar su cumplimiento [8]. En este caso se incorporó a un líder de proyecto con experiencia de 20 años en el

mantenimiento de un conjunto de sistemas de altas prestaciones. Este profesional ayudó a la elaboración de la guía a la vez que tomó el rol de Analista de RNF en los Sprints.

Etapa de Sprint Inicial. En esta etapa se incluyó la actividad de: "Identificar y clasificar los RNF". Esto consistió en la:

- Obtención de requerimientos preliminares: en esta actividad, se recopilan los requerimientos iniciales del cliente mediante entrevistas o reuniones.
- Identificación del tipo de aplicación software resultante. Existen diferentes tipos de software, como aplicaciones web, móviles, empresariales, entre otras.
- Identificación de problemas: se prepara una lista de preguntas para extraer los requerimientos del usuario con el fin de encontrar los problemas y los RNF relacionados.
- Clasificación del RNF: que se realiza en base al esquema de clasificación propuesto por Chung y do Prado Leite [9].
- Construcción y refinamiento interno de las historias de sistema: después de estimar los RNF candidatos el Analista de RNF construye las Historias de sistema, descritas más adelante.
- Presentación y refinamiento con el cliente: a partir de una breve reunión de trabajo se validan las Historias de sistema con el cliente.

Después de completar todos los pasos, se obtienen los RNF establecidos para su uso posterior junto con los RF.

Las Historias de sistema, tomadas de [10] son artefactos complementarios a las Historias de usuario con las que encapsular y manipular las características del comportamiento no funcional analizado en la sección anterior. Su estructura se muestra en la Tabla 1.

Tabla 1. Componentes de una Historia de sistema.

Componente	Descripción
ID RNF	Identificador del requerimiento no funcional.
ID HU	Identificador de la/s Historia/s de usuario del RF al cual está asociado este RNF.
RNF	Descripción del RNF.
Sub-RNF	Descripción de la subclasificación del RNF.
Prioridad	Prioridad del RNF asignada por el cliente.
Dependencia	RNF que depende del RNF actual.

El ID HU y el ID RNF vinculan ambos tipos de requerimientos, así por ejemplo el RNF: "Toda transacción debe realizarse en menos de un segundo" se relaciona con todas las HU cuya funcionalidad sea cumplida mediante una transacción.

La descripción del RNF y Sub-RNF incluyen el nombre de la propiedad, el tipo de propiedad y el valor, por ejemplo: "El tiempo de respuesta en el registro de búsqueda no debe ser mayor de 5 segundos." En este caso la descripción estructurada sería:

- RNF = Tiempo de respuesta
- Sub-RNF = Desempeño, Tiempo, comportamiento
- Tipo = Tiempo en segundos Valor = 5

La prioridad de los RNF es un valor de importancia relativa que considera su relevancia para el éxito general del proyecto. Para definir este puntaje se utilizó el modelo propuesto por Muhammad et al. [11]

Etapa de Ejecución del Sprint. Las Historias de usuario en conjunto con las Historias de sistema forman parte del *Product Backlog* y se emplean como insumo para detallar las Historias que se detallarán para ser tratadas en el Sprint. En esta etapa se incluyen en las Historia de usuario los criterios de aceptación, claros y cuantificables. Esto implica definir medidas específicas que se deben cumplir. Junto con ello será necesario definir una persona del equipo responsable de revisar la realización de la Historia de usuario [12]. Para el caso de la guía lo deja abierto a los miembros del equipo que se auto asignan la tarea.

De la misma manera, cada tarea del Sprint Backlog será documentada incluyendo criterios de aceptación como parte de la "definición de hecho", tal y como lo describe Behutiye en [12].

Fase 2. Una vez elaborada la guía se realizó la capacitación inicial a los equipos de trabajo. En marzo y abril de 2.022 iniciaron dos proyectos de desarrollo de aplicación web en tecnología .Net y PHP respectivamente. La empresa trabaja con la herramienta Jira para la gestión de tareas, clasificándolas por tipo y criticidad. Uno de los tipos de tareas en la herramienta son los "requerimientos" como paso previo a las Historias de usuario. Para el estudio de caso se tomaron los requerimientos, historias de usuario y tareas de Sprint de los últimos dos proyectos finalizados por cada equipo. Se analizó el contenido para describir los RNF detectados en base a la guía, se cuantificó la cantidad por tipo, si contenía los suficientes detalles para verificar el cumplimiento del RNF y su capacidad para ser probado. Este análisis fue realizado de forma independiente por dos de los autores del trabajo, comparando y discutiendo los resultados en caso de conclusiones divergentes.

De la misma manera se tomaron los RNF y las Historia de sistema descritos en los proyectos llevados adelante teniendo en cuenta la guía descrita en Fase 1.

Fase 3. Al finalizar cada proyecto la empresa realiza una encuesta de satisfacción siguiendo buenas prácticas de normativas para la gestión de calidad en el proceso de trabajo, como la ISO 9001. Para el estudio de caso se tomaron los resultados de las encuestas de los proyectos antes y después de la inclusión de la guía.

Una vez entregada la primera versión de cada proyecto iniciaron las tareas de mantenimiento, las cuales también fueron gestionadas en la herramienta Jira. Durante la carga de estas tareas se las relacionaba con las Historias de usuario y de sistema relacionadas. En base a esta información se pudo evaluar la cantidad y criticidad de los incidentes en el sistema cuyo origen tuvo que ver con el tratamiento de los requerimientos no funcionales.

Fase 4. Finalmente, durante la fase 4 se analizó el grado de agilidad del equipo contestando las preguntas de "esencial" del cuestionario [6]. Esto se llevó adelante en dos momentos: antes del inicio de los proyectos donde se incorporó la guía de RNF y una vez finalizado los mismos. De esta manera era posible comparar los resultados, además de un análisis cualitativo de las opiniones de los encuestados.

Finalmente, uno de los autores del trabajo participó en las reuniones de retrospectiva de los proyectos incluyendo preguntas al equipo de trabajo y al analista de RNF para

identificar la utilidad de la guía. Los resultados de estas interacciones se comentan en la sección de Discusión.

3. Resultados

A continuación, en la Tabla 2 se describen los resultados obtenidos a partir del análisis de los RNF detectados antes y después de la inclusión de las actividades de la guía elaborada en la Fase 1 del estudio de caso. Además de la columna con los equipos (#E) y los proyectos (#P) está la columna donde se indica la cantidad de puntos historia final de cada proyecto (PI), el número de RNF detectados (#RNF), la densidad de puntos historia por RNF detectado (DPR), la cantidad de RNF cuantificados (Cuant.) y la cantidad de RNF probados (Prob.); en estos dos últimos casos también está el porcentaje sobre el total de RNF.

Tabla 2. Análisis de los RNF.

#E	#P	PI	#RNF	DPR	Cuant.	% Cuant.	Prob.	% Prob
1	1	122	13	9,4	8	62%	6	46%
1	2	87	10	8,7	5	50%	4	40%
2	3	99	15	6,6	9	60%	8	53%
2	4	135	12	11,3	10	83%	9	75%
1	5	130	27	4,8	24	89%	22	81%
2	6	117	31	3,8	29	94%	26	84%

En la Fig. 2 se resumen los datos tomando los cuatro primeros proyectos realizados sin la guía y los dos últimos proyectos con la guía. Ambos equipos mejoraron la capacidad de detección y descripción de RNF, tanto en cantidad como en calidad.

De igual manera, los datos obtenidos en la fase 3 al analizar las encuestas de los clientes son los siguientes: en la tabla 3 se resume el grado de satisfacción y en la tabla 4 se resume la cantidad y calidad de los incidentes relacionados con aspectos de RNF.

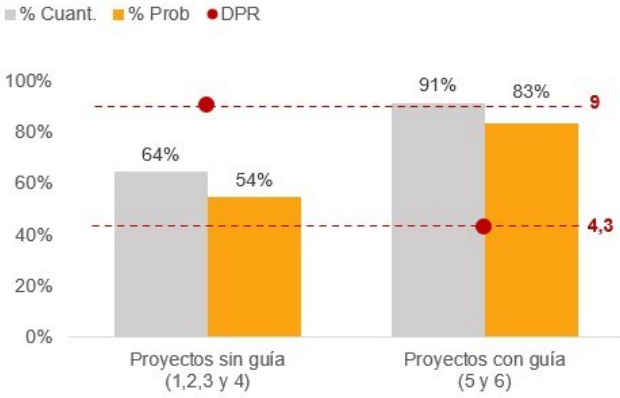


Fig. 2. Análisis de los RNF antes y después de incorporar la guía.

La tabla 3 muestra 7 de los ítems solicitados en la encuesta de satisfacción del cliente y las respuestas en escala Lickert de 5 puntos de los seis proyectos evaluados (de P1 a P6). En la columna PSG se promedian los cuatro primeros proyectos realizados sin la guía de RNF y en la columna PCG los dos proyectos desarrollados con la guía. Puede verse que la satisfacción del cliente fue similar con pequeñas variaciones en los ítems relacionados con las reuniones, la velocidad de entrega y la calidad percibida del cliente respecto de lo entregado.

Tabla 3. Grado de satisfacción del cliente.

Ítem	P1	P2	P3	P4	P5	P6	PSG	PCG
Las reuniones iniciales	4	4	4	4	5	4	4	4,5
Las reuniones de Planificación	4	4	4	4	4	5	4	4,5
Las reuniones de Demo	5	5	5	5	5	5	5	5
La velocidad de entrega de los desarrollos	4	4	5	4	4	4	4,25	4
La calidad de los desarrollos entregados	4	4	4	4	4	5	4	4,5
Confiaría en la empresa para futuros desarrollos	5	5	5	5	5	5	5	5
En general el servicio de la empresa fue:	5	5	5	5	5	5	5	5

En la tabla 4 se resume la información obtenida de las tareas de mantenimiento ingresadas en la herramienta de gestión de tareas: la cantidad de incidentes de mantenimiento por proyecto (#Incidentes), el promedio de criticidad que va del 1: criticidad muy baja al 5: criticidad muy alta (P. Criticidad) y la cantidad de horas totales invertidas en los incidentes (#Horas).

Tabla 4. Tiempo y criticidad de los incidentes relacionados con RNF.

#P	PI	#RNF	#Incidentes	P. Criticidad	# Horas	D_IPI	D_HI
1	122	13	8	3,8	58	15,3	7,3
2	87	10	9	3,8	55	9,7	6,1
3	99	15	7	4,9	74	14,1	10,6
4	135	12	4	3,3	23	33,8	5,8
5	130	27	5	2,6	28	26,0	5,6
6	117	31	3	2,7	18	39,0	6,0

Las dos últimas columnas son la densidad de incidentes por puntos historia (D_IPI) y la densidad de horas de trabajo por incidente (D_HI). Se puede ver una clara disminución de la cantidad de horas invertidas por incidente, menor criticidad y mayor densidad de funcionalidad entregada por incidente. Esto se aprecia agregando los datos antes y después de la implementación de la guía, tal y como se ve en la Fig. 3. Finalmente, durante la fase 4 se completó la tabla 5 tal y como se describió en la sección anterior. En la tabla solo figuran los ítems con alguna variación antes (columna A) y después (columna D) de implementar la guía. Estas dos columnas fueron completadas con tres opciones cerradas: SI, NO y Parcialmente (Parc.). También figura un ítem sin variación aparente pero donde se encontró alguna diferencia discutida a continuación de forma cualitativa.

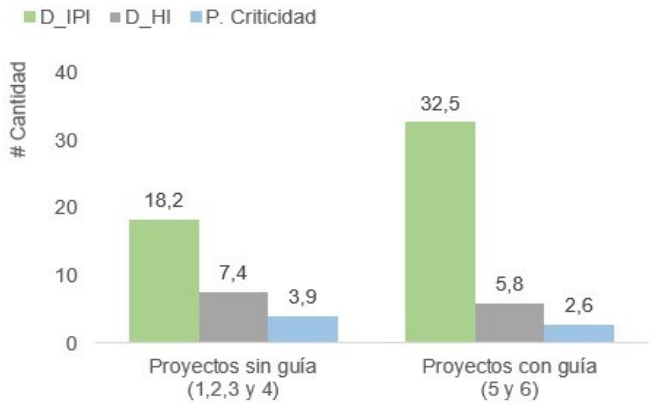


Fig. 3. Resumen de los incidentes antes y después de la implantación de la guía de trabajo con RNF.

El ingreso del Analista de RNF le ha quitado protagonismo al Dueño del Producto, en particular por el trabajo inicial y de sincronización que deben realizar entre las Historias de usuario y las Historias de sistema para organizar el *Product Backlog*. Respecto de las reuniones diarias, se ha trabajado de la misma manera solo que los problemas e impedimentos ahora también contemplan los relacionados con RNF. En

La duración del Sprint se vio aumentada, la inclusión de RNF aumentó la cantidad de tareas de pruebas y disminuyó un poco la velocidad del equipo. Finalmente, el producto al final de cada Sprint no siempre estaba funcionando, pero esto ya sucedía antes de la incorporación de la guía de RNF.

Tabla 5. Cuestionario interno de agilidad.

Ítems	**A**	D
El Dueño del Producto (PO) está empoderado para ordenar	SI	Parc.
PO tiene el conocimiento para ordenar el Product Backlog	SI	Parc
Problemas e impedimentos son mencionados en la reunión diaria	SI	SI
Longitud de iteración es de 4 semanas o menos	SI	Parc
Muestra producto funcionando y probado	Parc	Parc

4. Discusión

Esta sección discute los resultados obtenidos y responde a las preguntas de investigación.

PI1: ¿Hasta qué punto la guía facilitó la identificación temprana de los requerimientos no funcionales?

Para la primera pregunta, al realizarse el análisis de los RNF detectados se pudo encontrar una clara diferencia en cantidad y calidad. Por un lado, se pudieron incluir nuevos tipos de RNF a la vez que la plantilla de Historias de sistema mejoró la precisión

en las descripciones para que puedan ser probadas. La densidad de RNF respecto de los puntos historia fue de más del 100% tal y cómo se ve en la columna DPR de la tabla 3.

De la misma manera, los RNF fueron elaborados con mayor precisión lo cual llevó a realizar mayor cantidad de pruebas. Esto era esperable por la incorporación de un rol explícito trabajando para ello, junto con la guía y la capacitación previa de los equipos. La duración de la planificación por Sprint también aumentó en un 50%, pasando de 60 minutos a 90 minutos por semana planificada. Esto pudo suponer una disminución en la agilidad de los equipos que no se evidenció en estudios posteriores.

PI2: ¿Se ha observado una mejora en la satisfacción del cliente al utilizar la guía en el proceso de gestión de requerimientos no funcionales?

El grado de satisfacción del cliente no aumentó sustancialmente por el uso de la guía. De forma secundaria se pudo observar solo la mejora pequeña de los ítems relacionados con las reuniones y la comunicación, esto se puede justificar al aumentar la cantidad de interacciones entre el equipo y el cliente debido a la validación de las Historias de sistema. Las tareas de mantenimiento sí evidenciaron un cambio tanto en la cantidad, criticidad y número de horas invertidas en ellas. Esto aumentó la percepción de calidad del cliente respecto del producto software final, aunque no se reflejó totalmente en la encuesta de satisfacción por ser esta aplicada solo al final del proyecto y no en etapas posteriores. De la misma manera, el equipo de trabajo manifestó una mayor estabilidad en las tareas de mantenimiento, en particular por la disminución de los incidentes críticos, que como se observa en la tabla 5 y 6, disminuyeron casi a la mitad.

PI3: ¿La inclusión de nuevas prácticas y artefactos de trabajo afectó el grado de agilidad de los equipos de trabajo?

La agilidad del equipo no se vio modificada de manera significativa respecto de la encuesta llevada adelante o la percepción de los equipos de trabajo en las reuniones de retrospectiva. Un análisis cualitativo de las tareas después de la inclusión de la guía da cuenta de un aumento en la cantidad de tareas relacionadas con las Historias de sistema y las pruebas asociadas. Para este caso la respuesta de los equipos de trabajo fue aumentar la cantidad de tiempo de cada Sprint, intentando mantener estable el número de puntos historia entregados por iteración. Esta decisión de equipo puede ser vista como una disminución en la agilidad.

5. Conclusiones

El estudio presenta los resultados de un estudio de caso evaluando la inclusión de una guía para la gestión de requerimientos no funcionales en una pequeña empresa de desarrollo ágil de software.

Los datos para la evaluación de las preguntas de investigación se obtuvieron a partir del análisis de la herramienta de gestión de tareas de la empresa, la participación en las reuniones de retrospectiva de cada Sprint y la aplicación de instrumentos de recolección como las encuestas de satisfacción del cliente o el cuestionario para la medición de la agilidad de los equipos.

Como resultado de ello se observó una mayor cantidad y precisión al identificar los requerimientos no funcionales, su inclusión explícita en las discusiones de planificación

con el cliente y en el equipo de trabajo, llevando a una mayor cantidad de tareas en cada Sprint para su comprobación. Esto llevó a aumentar la estabilidad del proceso de desarrollo percibida por el equipo y, en menor medida, la calidad del producto final percibida por el cliente. La agilidad de los equipos no se vio disminuida, aunque aumento la duración de los Sprints como respuesta de los equipos para mantener la misma cantidad de funcionalidad entregada

Los resultados de este trabajo sirven como base para el desarrollo de procedimientos con los que gestionar requerimientos no funcionales en entornos ágiles de desarrollo y como primer paso para diseñar investigaciones posteriores en el marco de la metodología de Diseño de la Ciencia.

Agradecimientos. Este trabajo se realizó en el marco de los proyectos 21F001 y 21 F005 de la Secretaría General de Ciencia y Técnica de la Universidad Nacional del Nordeste

Referencias

1. Al-Saqqa, S., Samer S., and Hiba A.:Agile software development: Methodologies and trends, International Journal of Interactive Mobile Technologies 14.11 (2020).
2. Sommerville, Ian. Engineering software products. Vol. 355. London, UK: Pearson, 2020.
3. Rahy, S, and Bass J. Managing non-functional requirements in agile software development" IET software 16.1: pp. 60-72 (2022).
4. Behutiye, W., Karhapää, P., Costal, D., Oivo, M., & Franch: Non-functional requirements documentation in agile software development: challenges and solution proposal, nternational conference on product-focused software process improvement. Cham: Springer International Publishing, (2017).
5. Yin, R. Case study research: Design and methods pp: 687-704 (2003).
6. Kniberg, H.: Scrum checklist. vol,2, pp:1-2 (2010).
7. Van Den Broek, R., Bonsangue, M. M., Chaudron, M., & Van Merode, H.: Integrating testing into Agile software development processes. In 2014 MODELSWARD pp. 561574 (2014)
8. Bourimi, M., Barth, T., Haake, J. M., Ueberschär, B., & Kesdogan, D AFFINE for enforcing earlier consideration of NFRs and human factors when building sociotechnical systems following agile methodologies.Human-Centred Software Engineering: Third International Conference, HCSE (2010).
9. ChungJ L. Leite D: On non-functional requirements in software engineering, in Conceptual modeling: Foundations and applications, A. T. Borgida, V. K. Chaudhri, P. Giorgini, and E. S. Yu, Eds. Berlin Heidelberg: Springer pp. 363-379 (2009).
10. Cleland-Huang J, Vierhause M., 'Discovering, analyzing, and managing safety stories in agile projects,'' in Proc. IEEE 26th Int. Requirements Eng. Conf. (RE), pp. 262–273 (2018)
11. Muhammad, A. Siddique, A, Mubasher, M, Amjad, A., Quadri, N: Prioritizing Non-
12. Functional Requirements in Agile Process Using Multi Criteria Decision Making Analysis IEEE Access, 2023, vol. 11, pp. 24631-24654. (2023).
13. Behutiye, W., Rodríguez, P., & Oivo, M.: Quality Requirement Documentation Guidelines for Agile Software Development. IEEE Access, 10, pp: 70154-70173 (2022).

Estudios de casos realizados en PyMEs de Argentina para validar un conjunto de métricas para el despliegue de sistemas de software

Pablo Vázquez[1] [0009-0003-0528-4874], Marisa Panizzi[1] [0000-0002-9699-1544], Rodolfo Bertone[2] [0000-0003-0609-0310]

[1] Programa de Maestría en Ingeniería en Sistemas de Información. Escuela de Posgrado. Universidad Tecnológica Nacional. Regional Buenos Aires (UTN-FRBA). Medrano 951. CABA, Argentina.

[2] Facultad de Informática -Instituto de Investigación en Informática LIDI (III-LIDI). Universidad Nacional de La Plata, 50 y 120. La Plata, Argentina.

vazpablo@gmail.com, mpanizzi@frba.utn.edu.ar, pbertone@lidi.unlp.edu.ar

Resumen. La creciente demanda de productos software de calidad, exige a las desarrolladoras de software, el uso de estándares y metodologías que aseguren la entrega de dichos productos software con la calidad requerida. El despliegue es el proceso de software. por el cual se hace la transferencia del sistema de software a la empresa cliente Para las PyMES es fundamental mejorar sus procesos para ser competitivas. Es necesario que lleven adelante iniciativas que contribuyan con el desarrollo y mejora de su competitividad. El uso de métricas en el proceso de despliegue permite llevar el control de todas las actividades y tareas, reducir costos por retrabajo, evitar retrasos en la entrega del producto software, y aumentar la calidad del proceso. En este artículo se presentan los resultados de la validación de las métricas diseñadas para el proceso de despliegue de sistemas de software mediante dos estudios de casos desarrollados en PyMES de Argentina.

Palabras clave: métricas, proceso de despliegue, sistemas de software, estudios de casos, PyMES de Argentina.

1. Introducción

El despliegue de sistemas de software es un proceso crucial del ciclo de vida de desarrollo de software porque después de un despliegue exitoso, el sistema de software finalmente está operativo para que el cliente pueda beneficiarse económicamente de su uso [1].

El proceso de despliegue contiene prácticas que tienden a presentar problemas, como falta de componentes (en general externos), descargas incompletas e instalaciones erróneas, retrabajo por falta de capacidades y competencias tecnológica tanto de los recursos informáticos como los usuarios finales [2]. Los problemas que pueden ocurrir en la fase de despliegue se transfieren y eventualmente resuelven como parte de la fase de mantenimiento, por lo tanto, contar con un despliegue eficiente de software se debería ahorrar recursos en términos de costo y esfuerzo [1].

Con la aparición de un gran número de empresas de software transformó a la industria del software haciéndola más competitiva. La mejora en la calidad del software se ha convertido en un factor clave de éxito que permite a las empresas lograr una

ventaja competitiva y es esencial para la calidad del producto, la sostenibilidad comercial y el crecimiento [3].

En Argentina, la industria del software se compone mayoritariamente por PyMES, representando casi el 80% del sector [4] y requieren procesos sistematizados y de calidad que contribuyan con el desarrollo y mejora de competitividad frente al mercado.

En consecuencia, dada la relevancia del proceso de despliegue, los inconvenientes que conlleva un despliegue inadecuado más la necesidad de las PyMES de disponer de procesos sistematizados y de calidad para la entrega de sistemas de software que satisfagan las necesidades de sus clientes, llevo a definir un conjunto de métricas para el proceso de despliegue de sistemas de software en PyMES de Argentina

Para el diseño de las métricas se siguieron los siguientes pasos: el estado del arte mediante un mapeo sistemático de la literatura [5] y un estudio de caso desarrollado en una PyME de Argentina para obtener retroalimentación, por parte de profesionales de la industria con el fin de refinar y completar el conjunto de métricas [6].

En este artículo se presentan los resultados de la validación de las métricas diseñadas para el proceso de despliegue de sistemas de software mediante dos estudios de casos desarrollados en PyMES de Argentina.

El artículo se estructura de la siguiente manera: en la Sección 2 se presenta una descripción general del diseño de las métricas, en la Sección 3 se describen los estudios de casos. En la Sección 4 se presenta un análisis de las amenazas a la validez y, finalmente, en la Sección 5 se exponen las conclusiones y trabajos futuros.

2. Descripción general del diseño de las métricas

Para la definición de las métricas, se contemplaron las actividades y tareas del "proceso de transición" (en inglés, *Transtition Process*) del estándar ISO/IEC/IEEE 12207:2017 [7] por ser un estándar reconocido internacionalmente.

Se utilizó la clasificación de métricas propuesta por la familia del estándar ISO/IEC 9126 [8],[9],[10],[11], según su naturaleza: métricas básicas, métricas agregadas y métricas derivadas.

Para organizar el conjunto de métricas definidas para el proceso de despliegue de sistemas de software, se consideró una visión de métricas basada en tres dimensiones, también denominada "visión tridimensional" del proceso de despliegue [12], compuesta por las dimensiones "Proceso", "Producto" y "Persona". La dimensión ¨Proceso¨ dado que resultan de interés las fases o etapas, actividades y tareas que lo componen. La segunda dimensión, ¨Producto¨, contempla las características del producto de software como la complejidad de instalación, los requisitos de instalación, la integración con la infraestructura del cliente y el tamaño, entre otras. Y finalmente, la última dimensión ¨Persona¨, se la considera debido a la existencia del peopleware y su impacto en el proceso de despliegue de sistemas de software. Por restricciones de espacio, las métricas diseñadas para cada una de las dimensiones se encuentran en un anexo [13].

3. Estudios de casos

En esta sección, se presentan dos estudios de casos (EC1, EC2) realizados en PyMES desarrolladoras de software de Argentina con el propósito de validar la aplicabilidad del conjunto de métricas diseñadas con el propósito de completarlas y refinarlas si fuera necesario. Para la realización y reporte de los estudios de casos se siguieron las directrices propuestas por Runeson *et al.* [14].

3.1 Objetivos y preguntas de investigación (PI) de los estudios de casos

En la Tabla 1 se presenta el objetivo de los estudios de casos siguiendo el paradigma *Goal-Question-Metric* de Basili *et al.* [15].

Analizar	la documentación del despliegue de sistemas de software.
con el propósito de	evaluar la aplicabilidad de las métricas,
con respecto a	a las dimensiones “Proceso”, “Producto” y Persona”.
desde el punto de vista	de profesionales de la industria del software,
en el contexto de	PyMES de Argentina.

Tabla 1. Objetivo de los estudios de casos.

Para alcanzar este objetivo se plantearon las siguientes preguntas de investigación (PI):

PI1: ¿Que mediciones se llevaron a cabo durante las actividades del proceso de despliegue de sistemas de software (identificación y análisis)?

A través de esta pregunta, se busca obtener la información de las métricas que se recolectaron durante la ejecución del proceso de despliegue y el uso dado por la consultora para compararlo con la propuesta realizada.

PI2: ¿De qué manera se puede fortalecer el proceso de despliegue de sistemas de software en esta empresa?

Con esta pregunta se intenta determinar la forma en que la consultora puede fortalecer su proceso de despliegue, para esto se propone un conjunto de métricas identificando su objetivo y la forma de cálculo.

3.2 Contexto, caso y unidad de análisis

De acuerdo con la clasificación de Yin [16] los dos estudios de casos son holísticos y de casos únicos caracterizados por los elementos que se presentan en la Tabla 2. Se utilizó la clasificación de PyMES en Argentina según la cantidad de empleados [17].

Tabla 2. Contexto, caso y unidad de análisis de los estudios de casos (EC1, EC2).

Estudios de casos	Contexto	Caso	Unidad de análisis
EC1	**PyME Mediana de tramo 2** desarrolladora de software sistemas radicada en la Ciudad Autónoma de Buenos Aires, la misma cuenta con un plantel de 430 empleados.	Despliegue de un Portal de Recursos Humanos realizado en una entidad bancaria de la República Argentina.	Documentación del despliegue de un Portal de Recursos Humanos.
EC2	**PyME Mediana tramo 1** de sistemas radicada en la Ciudad Autónoma de Buenos Aires, la misma cuenta con un plantel de 60 empleados.	Despliegue de un Sistema de Gestión para una agencia de publicidad de la República Argentina.	Documentación del despliegue de un Sistema de Gestión.

3.3 Preparación para la recolección de los datos

En los estudios de casos participaron los dos jefes de proyectos de cada una de las PyMEs.

Las actividades de formación a los dos jefes de proyectos se realizaron de manera virtual vía la plataforma *Meet.* La duración de la formación en cada empresa ha sido de 1 sesión de 3 horas. Una vez finalizada la formación se mantuvo contacto permanente con la empresa vía telefónica, *WhatsApp* y por videoconferencia para la resolución de dudas que surgieron durante la aplicación de las métricas por parte de los jefes de proyectos durante un mes aproximadamente. Finalizado el despliegue, los jefes de proyectos de cada uno de los estudios de caso facilitaron la documentación generada durante dicho proceso para analizar la aplicación de las métricas propuestas.

Para la recolección de los datos se utilizó una técnica de tercer grado combinada con un método independiente según la clasificación propuesta en [18].

Se utilizó una plantilla con un esquema de codificación compuesto de 3 grupos (A1 Preparación del despliegue, A2 Realización del despliegue y A3 Gestión de los resultados del despliegue), cada uno de ellos coinciden con las 3 actividades del proceso técnico "Transición" del estándar ISO/IEC/IEEE 12207:2017 [7]. Para recolectar la información sobre las métricas utilizadas, se utilizó la codificación propuesta en la sección 2. En la Tabla 3 y en la Tabla 4 se presenta la trazabilidad de la documentación analizada de cada uno de los estudios de casos utilizando las plantillas diseñadas para la recolección de los datos.

Tabla 3. Trazabilidad de la documentación analizada para el EC1.

Documentos / Actividades	A1	A2	A3
Plan de trabajo	CAP, HEP, PAP PEP, ET, TP, ESF		
Informe de avance		%DesvioCap %DesvioHE %DesvioPEnt %AvanceE %AvancePA	

		%DesvioPA %DesvioEsf %AvanceTR %DesvioTR %DensidadD	
Informe de cierre			%DesvioCap %DesvioHE %DesvioPEnt %AvanceE %AvancePA %DesvioPA %DesvioEsf %AvanceTR %DesvioTR %DensidadD
Casos de Prueba del despliegue		CantDef %DensidadD	

Tabla 4. Trazabilidad de la documentación analizada para el EC2.

Documentos/ Actividades	A1	A2	A3
Plan de despliegue	CAP, HEP, PAP PEP, ET, TP ESF, E, VELH		
Informe de avance		%DesvíoCap %DesvíoHE %DesvíoPEnt %AvanceE %AvancePA %DesvíoPA %DesvíoEsf %AvanceTR %DesvíoTR %DensidadD PRD PRDT CAPR	
Cierre del despliegue			%DesvíoCap %DesvioHE %DesvioPEínt %AvanceE %AvancePA %DesvíoPA %DesvíoEsf %AvanceTR %DesvíoTR %DensidadD PRD PRDT VEL
Plan de Capacitación	CAP		
Casos de prueba	PA	PAE	PAE
Informe de ejecución de pruebas del despliegue		CantDef %DensidadD	

3.4 Análisis e interpretación de los resultados

En esta sección, en primer lugar, se presentan los resultados obtenidos del EC1 y EC2 sobre la aplicación de las métricas para dar respuesta a la PI1 (Ver Tabla 5). En segundo lugar, se describen las propuestas de mejora (informes entregados a las PyMEs) que permitió dar respuesta a la PI2 (ver Tabla 6).

Tabla 5. Resultados obtenidos para dar respuesta a la PI1 (EC1 y EC2).

Estudio de caso	Actividad 1. Preparación del Despliegue	Actividad 2. Ejecución del Despliegue	Actividad 3. Gestión de resultados del despliegue
EC1	• Planificación de las tareas a desempeñar. • Estimación del esfuerzo de cada una de las tareas. • Casos de test diseñados.	• Registro del avance de las tareas. • Casos de test ejecutados de forma incompleta. • Registro de defectos. • Ejecución de tareas no planificadas.	• Problemas con los repositorios de software. • Múltiples inconvenientes técnicos y del producto impactaron fuertemente en la calidad final del producto y la satisfacción de los usuarios.
EC2	• Planificación de las tareas a desempeñar. • Estimación del esfuerzo del despliegue y de cada una de las tareas. • Casos de prueba diseñados.	• Registro del avance y desvío de las tareas. • Casos de test ejecutados de forma inconsistentes. • Registro de defectos.	• Desvíos en la planificación. • Esfuerzo mayor al planificado. • Inconsistencia entre ambientes de QA y Producción.

Tabla 6. Informes con las propuestas de mejora en respuesta a la PI2.

Informe EC1	Informe EC2
• Planificar y estimar cada una de las fases del ciclo de vida de desarrollo del software. • Realizar mediciones en todas las fases del ciclo de vida del desarrollo de software. • Identificar las mediciones relevantes. • Elaborar el plan de pruebas de despliegue. • Identificar las capacidades necesarias para llevar adelante cada una de las actividades del despliegue. • Identificar los riesgos del proceso.	• Las mediciones de software son el mejor método de control y contribuyen a la toma temprana de decisiones. • Definir de manera clara y precisa las actividades a realizar desde el inicio. • Ambientes de calidad parecidos a los de producción. • Planificar y estimar para mostrar avances constantes que den valor al cliente. • Identificar las mediciones relevantes para cada fase del ciclo de vida de desarrollo del software. • Identificar los riesgos del proceso.

4. Amenazas a la validez

Para analizar la validez de los estudios EC1 y EC2 se tuvieron en cuenta los factores propuestos en [14]:

- Validez de constructo. Los resultados se obtuvieron en base al análisis documental realizado y de la aplicación del conjunto de métricas para el proceso de despliegue de sistemas de software en dos contextos reales, lo que nos permitió responder a las preguntas de investigación definidas, determinando su pertinencia e idoneidad para el caso.
- Validez interna. La documentación utilizada pertenece a dos casos reales de la República Argentina. Para lograr una mayor precisión y validez del proceso estudiado, se reconoce la necesidad de combinar la fuente de datos (documentación del proyecto) con otro tipo de fuente, como entrevistas y / o grupos focales para garantizar una "triangulación de datos (fuente)". Además, los datos cualitativos recopilados y analizados podrían combinarse con datos cuantitativos resultantes del proyecto, asegurando así una "Triangulación Metodológica".
- Validez externa. El uso de solo dos estudios de caso puede limitar la generalización de los resultados. Sin embargo, se realizó un estudio preliminar en [6] y en este caso se considera necesario informar sobre estos hallazgos, ya que sirve como un incentivo para que otros investigadores repitan nuestro estudio en diferentes estudios de casos.
- Fiabilidad. Los datos de EC1 y EC2 fueron recopilados por un solo investigador. Aunque fueron analizados con los directores de tesis, esto puede considerarse como una amenaza para la investigación. Para agregar un mayor grado de confiabilidad, sería aconsejable que otro investigador aplique la plantilla con la codificación diseñada en otros estudios de casos.

5. Conclusiones y trabajos futuros

Se presentaron los resultados de EC1 y EC2 para determinar la viabilidad de la aplicación de un conjunto de métricas para el proceso de despliegue de sistemas de software en un entorno real.

A continuación, se detallan las conclusiones obtenidas después de llevar a cabo los estudios de caso:

- Se constató, que los profesionales que participaron en los estudios de casos consideran que las métricas propuestas impactan positivamente en la mejora de la calidad de los despliegues de los sistemas de software de sus PyMES.
- Como fortaleza de las métricas propuestas se encuentra la flexibilidad que aporta al poder aplicarse las justas y necesarias de acuerdo con las actividades y tareas a realizarse durante el despliegue, permitiendo a las PyMES llevar a cabo sus despliegues de sistemas de software desarrollados de manera sistematizada y controlada, adecuándose a sus recursos y necesidades.
- Los estudios de casos permitieron identificar a través del análisis documental falencias en el uso de métricas, ya que algunas de las mismas no lograron aplicarse en ninguno de los estudios de casos.
- También nos permitió elaborar una serie de recomendaciones para que la empresa implemente y mejore su proceso de despliegue, así como para introducir el uso de métricas para futuros despliegues de sistemas de software. Además, se sugirió el

uso de métricas para otros procesos del ciclo de vida de desarrollo de software.

Teniendo en cuenta los resultados obtenidos en los estudios de caso, como futuros trabajos se identifican: (a) aplicar las métricas a otros tipos de sistemas de información y recolectar información mediante una encuesta basadas en TAM (*Technology Acceptance Model*), con el objetivo de analizar el nivel de aceptación y de utilidad de la aplicación para el proceso de despliegue de sistemas de software, (b) aplicar las métricas en otros tipos de empresas, como por ejemplo empresas emergentes (Startups) o empresas de mayor tamaño para recolectar feedback por parte de los usuarios y contrastarlo con los resultados obtenidos en PyMES, (c) proponer nuevas métricas basadas en indicadores obtenidos a partir del diseño de encuestas de satisfacción.

Referencias

1. Subramanian, N. (2017). The software deployment process and automation. CrossTalk, 30(2):28-34.
2. Jansen S., Brinkkemper S. (2006). Definition and validation of the key process of release, delivery and deployment for product software vendors: Turning the ugly duckling into a swan IEEE International Conference on Software Maintenance, ICSM, art. no. 4021334, pp. 166-175.
3. Abushama, H. (2016). PAM-SMEs: process assessment method for small to medium enterprises. Software: Evolution and Process, 28(8):689 –711.
4. OPPSI-CESSI. (27 de 04 de 2019). Reporte anual del sector de software y servicios informáticos de la República Argentina del año 2018. Ciudad Autónoma de Buenos Aires: Cámara de la Industria Argentina de Software (CESSI).
5. Vázquez P., Panizzi M. y Bertone R. (2021). Métricas para el proceso de despliegue de
6. sistemas software: Un mapeo sistemático. Desarrollo e Innovación en Ingeniería. Vol. II, ed. 6, pp. 386-396. ISBN 978-958-53278-6-3.
7. Vázquez P., Panizzi M., Bertone R. (2021). Refinamiento de métricas para el proceso de despliegue de sistemas de software: Estudio de caso. In Proceedings of the IADIS Ibero American Conference Applied Computing (CIACA 2021). Del 18 al 19 de noviembre de 2021. ISBN 978-989-8704-35-1.
8. ISO/IEC/IEEE 12207:2017(E). (2017). Systems and software engineering — Software life cycle processes.
9. ISO/IEC 9126-1. (2001). Software engineering - Product quality - Part 1 Quality model.
10. http://www.iso.org/iso/home.html.
11. ISO/IEC 9126-2. (2003). Software engineering - Product quality - Part 2 External metrics.
12. http://www.iso.org/iso/home.html.
13. ISO/IEC 9126-3. (2003). Software engineering - Product quality - Part 3 Internal metrics.
14. http://www.iso.org/iso/home.html.
15. ISO/IEC 9126-4. (2004). Software engineering - Product quality - Part 4 Quality in use metrics. http://www.iso.org/iso/home.html.
16. Panizzi M., Davila M., Hodes A., Vázquez P., Ortiz F., Arana F., Bertone R. (2020). Desafíos para la implantación de sistemas de software. En las Actas del XXII Workshop de Investigadores en Ciencias de la Computación (WICC 2020), El Calafate, Argentina 7 y 8 de mayo de 2020. ISBN 978-987-3714-82-5.
17. Pablo Vázquez, Marisa Panizzi. Rodolfo Bertone (2023). Anexo - Estudios de casos realizados en PyMEs de Argentina para validar un conjunto de métricas para el despliegue de sistemas de software. Disponible en: https://doi.org/10.6084/m9.figshare.23826096

18. Runeson P, Höst M, Rainer A, Regnell B. (2012). Case study research in software engineering: guidelines and examples. Wiley Publishing, Hoboken.
19. Basili, V. and Rombach, D. (1988). The TAME project: towards improvement-oriented software environments. IEEE Transactions on Software Engineering, 14(6):758-773.
20. Yin, R.K. (2014). Case study research: design and methods. 5th Edition. Sage Publications.
21. Ministerio de Desarrollo Productivo. (2018), "Nuevas categorías para ser PyMES". https://www.argentina.gob.ar/noticias/nuevas-categorias-para-ser-pyme.
22. Lethbridge T., Sim S., Singer J. (2005). Studying software engineers: data collection techniques for software field studies. Empir Softw Eng 10(3), pp. 311–341.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Identificación Top-Down de Variedad de Contexto: Un Caso de Estudio en Fluctuaciones de la Napa Freática*

Mariano Campetella[1], Alejandra Cechich[1], Agustina Buccella[1], Ayelén Montenegro[2], Ángel Muñoz[2], and Andrea Rodríguez[2]

[1] GIISCO Research Group
Departamento de Ingeniería de Sistemas - Facultad de Informática
Universidad Nacional del Comahue, Neuquen, Argentina
mariano.campetella@est.fi.uncoma.edu.ar,
alejandra.cechich,agustina.buccella}@fi.uncoma.edu.ar

[2] Instituto Nacional de Tecnología Agropecuaria (INTA)
Alto Valle de Río Negro y Neuquén
montenegro.ayelen@inta.gob.ar

Abstract. Considerando la cantidad y diversidad en los datos que hoy día se relevan para futuros análisis, su combinación y uso se torna un elemento complejo a modelar. Es por esto que el agregado de semántica, a través de modelos conceptuales, es una tendencia actual en las arquitecturas software para Sistemas Big Data. En ese sentido, en este artículo presentamos una caracterización de contexto mediante la identificación top-down de variedad en sistemas predictivos sobre fluctuaciones de cuerpos de aguas subterráneos. Esa caracterización favorecería la identificación de situaciones recurrentes, incluyendo la posibilidad de reusabilidad durante el análisis. La propuesta se ejemplifica mediante dos casos comparativos en zonas geográficas diferentes y distantes.

Keywords: Reusabilidad de Software · Variedad en Sistemas Big Data · Taxonomías

1. Introducción

Nuestra propuesta para modelar variedad en sistemas big data (SBDs) se ve influenciada por distintos tipos de diversidad [1]: de *fuentes*, detectando distintas estructuras en los datos, técnicas de adquisición, etc.; de *contenido*, enfocando en las variables que son relevantes a un determinado problema de dominio; y de *procesamiento*, detectando variaciones en técnicas de análisis. Sin embargo, la variedad de los datos ha sido también considerada desde el punto de vista de incorporar semántica al proceso de modelado, incluso relacionando propiedades de calidad como interoperabilidad, seguridad, reusabilidad, etc. Entonces, en trabajos previos, hemos redefinido y extendido la clasificación de variedad para incluir la *variedad de contexto* [9], definiéndola como aquella que se refiere a *variaciones de dominio que pueden restringir o afectar los resultados del análisis*. Los diferentes tipos de contexto dependen de cada dominio; por ejemplo, para sistemas hidrológicos, los cuerpos de agua pueden clasificarse en ríos, lagos, mares, etc., que pueden verse influenciados por

* Este trabajo esta parcialmente soportado por el proyecto UNCOMA 04/F019 “Variedad en Sistemas Big Data” 2022-2025.

la climatología, geología, u otros factores de entorno. Por lo tanto, estructurar conocimiento de dominio es una tarea esencial para detectar la variedad contextual.

El proceso de identificación de variedad en SBDs puede iniciarse desde la evidencia que presenten los datos en datasets (enfoque bottom-up); desde el análisis conceptual del dominio (enfoque top-down); o con una mezcla de ambos casos. En [9] esos enfoques se discuten con cada uno de sus pasos, incluyendo la caracterización de dominio que deriva en la construcción de artefactos mediante los cuales se resguardan variables y relaciones relevantes para describir contextos similares. Esos artefactos son gestionados por una herramienta de soporte (CoVaMaT), que permite almacenarlos con una estructura que facilite su recuperación y reuso [10].

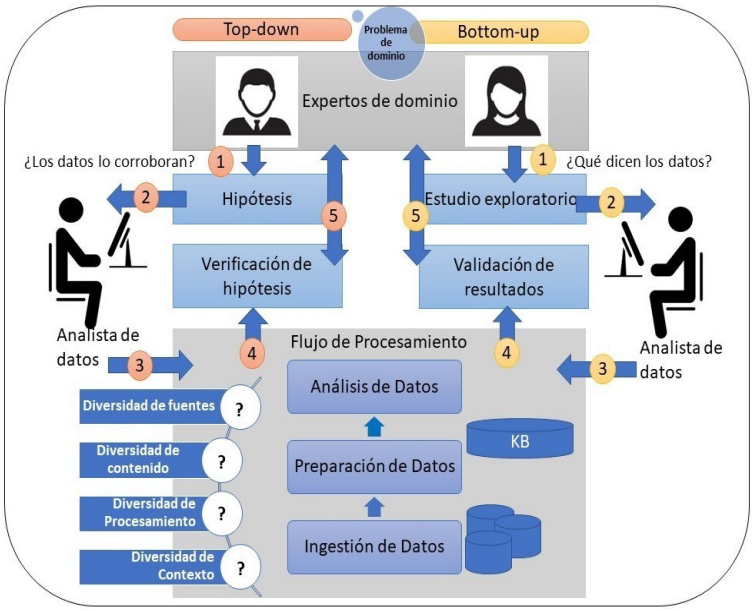


Fig.1. Enfoques T-VIP y B-VIP para identificación de variedad [9]

La Figura 1, muestra los dos enfoques paso a paso:

- Enfoque top-down (T-VIP): en este enfoque, a la izquierda de la figura, (1) dado un problema de dominio, el usuario experto elabora una o más hipótesis que deberán ser probadas mediante análisis de datos (¿los datos lo corroboran?); (2) luego, en base a esas hipótesis, analistas de datos proceden a realizar las tareas del flujo de procesamiento (3); finalmente, los resultados permiten verificar las hipótesis (4) posiblemente visualizando los datos de diferentes maneras, incluso permitiendo reformulaciones (5).
- Enfoque bottom-up (B-VIP): en este enfoque, a la derecha de la figura, (1) dado un problema de dominio, el usuario experto decide iniciar un estudio exploratorio para averiguar lo que los datos pueden revelar sobre ese problema (¿qué dicen los datos?); (2) luego, el estudio es llevado a cabo por analistas de datos aplicando las

actividades del flujo de procesamiento (3); finalmente, los datos retornan (4) para ser validados por expertos, finalizando el proceso o reiniciando la búsqueda (5).

La Figura 1 resalta activos en una base de conocimiento (KB). La forma en que esos artefactos se identifican varía dependiendo del enfoque. Por ejemplo, en el caso top-down, implica entender e investigar las características del dominio, probablemente estructurando el conocimiento como taxonomías, antes de proceder a contextualizar y almacenar cada caso de estudio para su reuso futuro. En este artículo, presentamos una caracterización top-down realizada para el reuso de sistemas predictivos sobre un dominio concreto: las fluctuaciones de napas freáticas influenciadas por condiciones meteorológicas y de entorno. Las variables relevantes del dominio son organizadas en forma de una taxonomía, que luego es instanciada en dos casos de estudio a modo comparativo para destacar la influencia de ese entorno en posibilidades de reuso de sistemas big data.

El artículo se organiza de la siguiente manera. La sección siguiente describe la caracterización de dominio realizada en el enfoque top-down para el problema bajo estudio. Luego, se analizan dos casos de estudio de manera comparativa y a modo de prueba de conceptos. Conclusiones y trabajos futuros se abordan al final.

2. Caracterización de influencias en la predicción de fluctuaciones de la napa freática

La Figura 2, muestra el proceso T-VIP para el caso de estudio comparativo. Note que se accede a la base de soporte gestionada por CoVaMaT para ingresar, consultar o reusar casos de variedad a modo de hojas funcionales* [9]. Las siguientes secciones, describen el proceso paso a paso.

2.1 T-VIP paso a paso: Identificación de variables

(1) El problema de dominio. Como puede verse en la figura, el problema de dominio se refiere al análisis de factores que se relacionan con fluctuaciones de cuerpos de agua subterráneos o napas freáticas. El uso de big data en este campo nos permite saber dónde ocurrirán las mayores fluctuaciones para brindar, a quienes deban tomar decisiones, una idea de qué áreas requerirán más atención y dónde deberían asignarse recursos más adecuadamente para prevenir daños y riesgos causados por cambios en los niveles del agua [5]. En este caso comparativo, la hipótesis planteada corresponde a analizar si *la variedad de contexto meteorológico y de entorno influye en el análisis de las fluctuaciones de la napa freática*. El primer caso corresponde a la zona de la cuenca mediterránea; mientras que el segundo se desarrolla en la zona del Alto Valle del Río Negro, en cooperación con expertos del Instituto Nacional de Tecnología Agropecuaria (INTA) Alto Valle.*

* Las hojas de datos funcionales (datasheets) permiten representar variedad de dominio y servicios que interactúan con la funcionalidad requerida.

* https://www.argentina.gob.ar/inta

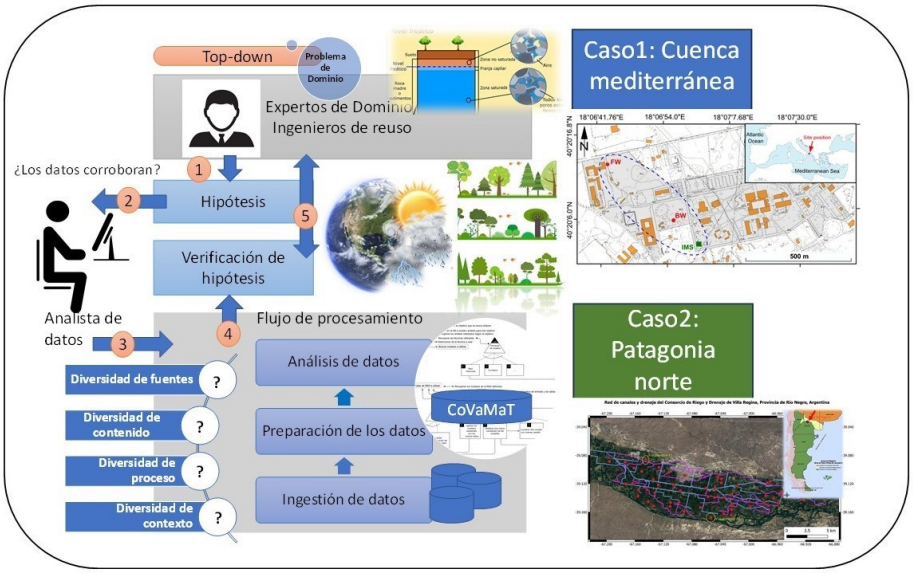


Fig.2. Enfoque top-down (T-VIP) para el dominio de estudio

(2) Análisis de las hipótesis. En esta actividad se requiere primero identificar las variables que influencian las fluctuaciones de la napa freática. Para abordar esta identificación, el ingeniero de reuso procede al *análisis de dominios*, considerando trabajos relacionados y aportes de expertos.

De acuerdo al análisis, los factores más comunes que afectan a las fluctuaciones de la napa freática, observadas en varios estudios, son las **lluvias** y las **temperaturas** [2,5–7,11,13]. Por ejemplo, en [5] se detectó una correlación positiva entre las variables de temperatura y las recargas de la napa freática. Lo mismo ocurre en [6], donde todos los sitios muestran un aumento en la recarga con el aumento de la temperatura y también con el aumento de las precipitaciones. Respecto a los estudios que analizan las precipitaciones, en particular la **nieve** como en [7,13], se llega a la conclusión de que es un componente que forma parte de la recarga de las aguas subterráneas. También se encontró que la **humedad** puede ser un determinante de la fluctuación y se puede observar en los estudios en [8,13]. Otra variable meteorológica que se tomó en cuenta para analizar las fluctuaciones es la **radiación solar** presente en los estudios en [5,6]. En el primero se puede observar que existe una correlación entre la temperatura con la radiación solar entrante, pero se muestra mayor correlación sobre la temperatura máxima. Por último, estudios que incluyen en sus análisis el **NDVI*** y la vegetación [4,12], mostraron resultados de correlación negativa entre el NDVI y los cambios en la precipitación y la profundidad del agua subterránea, también observando que el cambio de profundidad del agua subterránea está relacionado con las condiciones atmosféricas y superficiales, como la precipitación, la temperatura, la absorción del suelo, el contenido de agua de la vegetación, la humedad del suelo y la evaporación.

* El Índice de Vegetación de Diferencia Normalizada (NDVI) es un indicador simple de biomasa fotosintéticamente activa o, en términos simples, un cálculo de la salud de la vegetación.

Una taxonomía de influencias en fluctuaciones. El análisis del paso (2) identificó una serie de variables (Figura 3), que se conforma de cuatro grandes categorías: (1) Variables meteorológicas (2) Suelos (3) Tipo de vegetación y (4) Características estacionales.

Por ejemplo, la *precipitación* puede ser líquida o sólida. La primera incluye la lluvia y la llovizna; en la superficie de la tierra o en diferentes objetos, la precipitación líquida puede formarse como rocío o película líquida. La precipitación sólida incluye partículas de hielo suspendidas en la atmósfera que caen sobre la superficie terrestre. El *tipo de suelo* hace referencia a la textura del suelo, que es la proporción (en porcentaje de peso) de las partículas menores a 2 mm de diámetro (arena, arcilla y limo). Esto determina clases texturales, que se encuentran clasificadas en lo que se conoce como "triángulo textural"[**]. La *infiltración* se refiere al movimiento del agua en el suelo que ocurre después de la aplicación de riego o durante la lluvia, donde el agua ingresa verticalmente desde la superficie hacia el interior del suelo; y la *permeabilidad* mide la velocidad de esa infiltración. El *tipo de vegetación* se refiere a la capa de plantas que crece en un área específica, ya sea cultivada o de forma natural. La vegetación exhibe variaciones en cuanto a su tamaño, estructura y patrones estacionales, así como en los cambios que experimenta a lo largo del tiempo. Las *características estacionales* resumen aspectos que afectan el clima en períodos determinados y por diversas causas, ej. monzones o fenómeno de El Niño.

[**] R- 001- Guía para la determinación de textura de suelos por método organoléptico: http://www.prosap.gov.ar/



Fig.3. Taxonomía de variables relevantes en el análisis de las fluctuaciones

3. Análisis comparativo: instanciación de casos

El *Caso 1* se sitúa en la cuenca mediterránea, donde desde Junio 2017, el Instituto de Ciencias Atmosféricas y Clima del Centro Nacional de Investigaciones de Italia (ISAC-CNR*) ha tomado medidas de fluctuaciones de la napa de un acuífero en zona kárstica [3]. Las medidas fueron tomadas en dos freatímetros localizados en un área suburbana, dentro del campus de la universidad de Salento, con vegetación local mixta, incluyendo arbustos de tipo mediterráneo, pinos y olivos. El *Caso 2*, se situá en el norte de la Patagonia, en el Alto Valle del Río Negro cercano a la zona de la ciudad de Villa Regina, donde el INTA Alto Valle registra las fluctuaciones de la napa freática desde el año 2010. Las medidas corresponden a dos freatímetros situados en tipos de suelos y vegetación diferentes.

* https://www.isac.cnr.it/en/node/14864

3.1 T-VIP paso a paso: ¿los datos lo corroboran?

(3) Flujo de procesamiento. Para cada actividad, se muestra una síntesis de lo realizado, en ambos casos, por el analista de datos.

Ingestión de datos

- Caso 1: los datos provistos en el repositorio corresponden a mediciones de nivel freático, variables meteorológicas (temperatura, precipitación, evapotranspiración, humedad, etc.), variables de infiltración, tipo de suelo y vegetación, todas obtenidas de fuentes no estructuradas.
- Caso 2: los datos provistos por el INTA Alto Valle para trabajar provinieron de diversas fuentes (datasets estructurados, no estructurados, semi - estructurados, espaciales vector y espaciales raster). La Figura 4 muestra parte de esos datos, tomando como punto de variación a los tipos y como variaciones a los lotes de datos en sí mismos.

Archivo	Descripción	Tipo de fuente
grilla_suelos.shp	Capa que divide al suelo de la zona en cuadrantes donde de cada uno se tiene las características que lo identifican (geomorfología, permeabilidad, clase textural, etc).	Geoespacial - Vector (shp)
freatimetros_2010-2021.csv	Medición mensual del nivel freático de todos los freatímetros de la *red_freatimetros* desde el 2010 al 2021.	Estructurados (csv)
meteorologia_regina.txt	Datos producidos por la estación meteorológica de Villa	No estructurados (txt)
informes_meteorologia_regina.pdf	Regina cada 10 mi el 2009 hasta 2021. Resúmenes mensu	

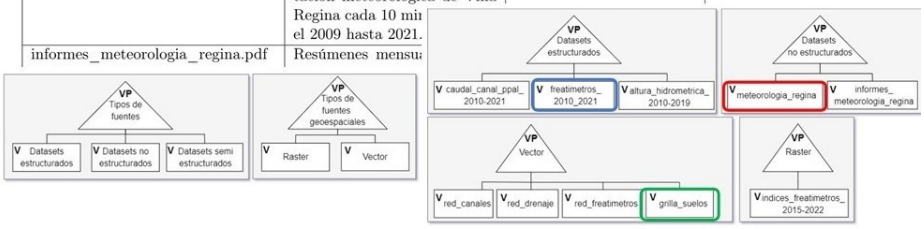
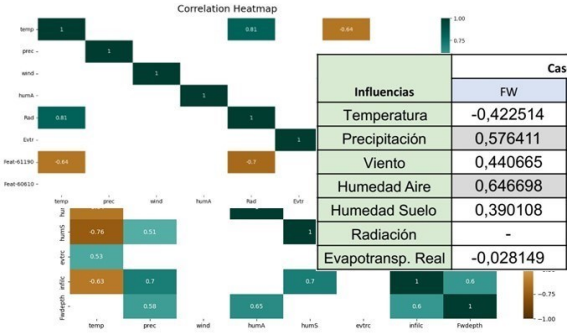


Fig.4. Variedad de las fuentes - Caso 2

Preparación de los datos. En ambos casos, la preparación consistió en conversión de datos no estructurados (pdf, txt) a formato csv. También en la identificación de campos nulos y su reemplazo analizando distintas técnicas; y en la reunión de fuentes de datos de distintos períodos de mediciones. Después del preprocesamiento, se analizó la correspondencia de variables disponibles con respecto a la taxonomía. Como resultado, ambos casos recolectaron datos de las variables meteorológicas, excepto por *humedad del suelo* (recolectado en el caso 1 pero no en el caso 2) y *radiación* (recolectado en el caso 2 pero no en el caso 1), lo que implica la detección de variedad de contenido entre ambos casos.

Análisis de los datos. La correlación, analizada con el coeficiente de Pearson para ambos casos, entre datos meteorológicos y niveles freáticos mostró (correlaciones en valor absoluto mayores a 0,5, Figura 5):

- Caso 1: En ambos pozos (FW y BW), las variables que muestran mayor influencia en las variaciones de la napa son: *precipitación* (FW, 0,576411; BW, 0,513294) y *humedad del aire* (FW, 0,646698; BW, 0,668482).
- Caso 2: Se detectaron discrepancias entre ambos freatímetros. Mientras que en el freatímetro 61190 la *temperatura* (-0,641544) y la *radiación* (-0,695296) se muestran como las mayores influencias; en el freatímetro 60610 no se observa ninguna correlación significativa.



Influencias	Caso 1		Caso 2	
	FW	BW	61190	60610
Temperatura	-0,422514	-0,368847	-0,641544	-0,261276
Precipitación	0,576411	0,513294	-0,083847	-0,111565
Viento	0,440665	0,477952	-0,006905	-0,081671
Humedad Aire	0,646698	0,668482	0,227759	0,05383
Humedad Suelo	0,390108	0,269569	-	-
Radiación	-	-	-0,695296	-0,2451
Evapotransp. Real	-0,028149	-0,086867	0,010203	-0,029608

Fig.5. Correlación meteorología-nivel de la napa freática

(4) Visualización de resultados y conclusiones del caso. Este primer análisis muestra que las variables meteorológicas influencian las variaciones en la napa freática de distintas maneras. Viendo la taxonomía, existen otras *variables de contexto* que podrían analizarse para dar respuestas a estas discrepancias. Por ejemplo, el Caso 1 corresponde a una zona boscosa del mediterráneo, con una precipitación media anual de 1,67 mm (2017-2022), con suelos de tipo fA (franco arenoso), permeabilidad muy rápida y pozos (FW,BW) ubicados en cercanía de zona urbana (parte superior de la Figura 6). La opinión experta es que es razonable esperar que en ese contexto, las variables meteorológicas más significativas sean las indicadas (el nivel de precipitaciones sumado a la vegetación hace que la humedad del aire sea relevante). Sin embargo, el Caso 2 corresponde a dos áreas de contextos diferentes: el freatímetro 61190 se ubica en la proximidad de una zona de meseta con suelos de tipo fl/f a fl (francolimoso/franco a francolimoso), con permeabilidad muy lenta y vegetación de tipo cultivos hortalizas; mientras que el 60610 se ubica próximo al Río Negro con suelos de tipo fA-Af/A-Af (francoarenoso a arenofranco/arenoso a areno franco) con permeabilidad muy rápida y vegetación de tipo pradera (parte inferior de la Figura 6). Nuevamente, el análisis con expertos revela que esas características pueden ser determinantes: el tipo de suelo y la zona cercana a la meseta hacen que las variaciones de la napa freática sean más dependientes de la recepción de agua en forma de precipitaciones y de su evaporación por radiación, aunque la humedad del aire no sea tan preponderante (precipitación media anual 0,67mm); mientras que la zona lindante al río, con un suelo permeable, registra variaciones en la napa de menor relevancia debido las precipitaciones y radiación, ya que la cercanía del cuerpo de agua es determinante. Finalmente, los resultados muestran que la hipótesis se corrobora a través de los datos; pero se puede ir más allá, sugiriendo cambios en el modelo de caracterización; por ejemplo, podría

agregarse la característica *proximidad de un cuerpo de agua*, para identificar el caso de zonas influenciadas por este contexto. Por último, los resultados almacenados en CoVaMaT permitirán la instanciación de futuros casos, reusando experiencias previas lo más cercanas a su contexto, para priorizar variables y métodos de análisis.

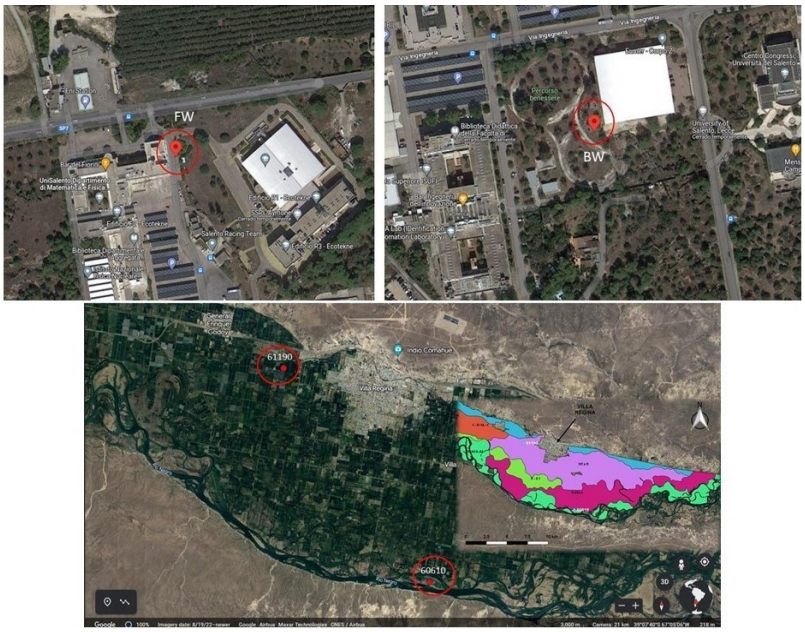


Fig.6. Variedad de contexto en ambos casos

4. Conclusiones y Trabajo Futuro

En este artículo hemos presentado el enfoque top down (T-VIP) para identificación de variedad en sistemas big data ejemplificado mediante un caso de estudio, donde la variedad de contexto es determinante para entender la relevancia de las características que definen las variables del caso de análisis. Actualmente, la investigación continúa incorporando nuevos casos y/o técnicas de análisis, a partir de la red de más de 50 freatímetros instalada en el Alto Valle. Esto permitirá seguir validando la taxonomía propuesta, como extender el modelo para evaluar el grado de reuso en cada situación.

References

1. Abawajy, J.: Comprehensive analysis of big data variety landscape. International Journal of Parallel, Emergent and Distributed Systems **30**(1), 5–14 (2015). https://doi.org/10.1080/17445760.2014.925548
2. Basant Yadav, Pankaj Kumar Gupta, N.P., Himanshu, S.K.: Ensemble modelling framework for groundwater level prediction in urban areas of india **712** (135539) (2020). https://doi.org/10.1016/j.scitotenv.2019.135539

3. Delle Rose, M., Martano, P.: Datasets of groundwater level and surface waterbudget in a central mediterranean site (21 june 2017ndash; 1 october 2022). Data **8**(2) (2023). https://doi.org/10.3390/data8020038
4. Ge Song, Jin-ting Huanga, B.h.N.J.w.W.L.Z.: Effects of groundwater level onvegetation in the arid area of western china. China Geology **3**, 527535 (2021). https://doi.org/10.31035/cg2021062
5. Gonzalez, R.Q.: Prediction of groundwater levels in a changing climate. Masterthesis, MSc Geoinformatics (2021)
6. J. L. McCallum, R. S. Crosbie, G.R.W.W.R.D.: Impacts of climate change ongroundwater in australia: a sensitivity analysis of recharge. Hydrogeology **18**, 1625–1638 (2010). https://doi.org/10.1007/s10040-010-0624-y
7. L.K. Mancewicz, L. Davisson, S.W.E.B.S.P., Tyler, S.: Impacts of climate changeon groundwater availability and spring flows: Observations from the highly productive medicine lake highlands/fall river springs aquifer system. AMERICAN WATER RESOURCES ASSOCIATION **57**(6), 1021–1036 (2021)
8. Mahnoosh Moghaddasi, Mehdi Mohammadi Ghaleni, M.A., Salari, S.: Potentialimpacts of climate change on groundwater levels in golpayegan plain, iran. Water Supply (2022)
9. Osycka, L., Buccella, A., Cechich, N.A.: Data variety modeling: A case of contextual diversity identification from a bottom-up perspective. In: 27th Argentine Congress, CACIC 2021, Salta, Argentina, October 4-8, 2021, Revised Selected Papers. Communications in Computer and Information Science 1584. pp. 124–138. Springer (2022)
10. Osycka, L., Cechich, A., Buccella, A., Montenegro, A., Muñoz, A.: Covamat: Functionality for variety reuse through a supporting tool. In: XI JCC-BD&ET (2023)
11. Safieh Javadinejad, Rebwar Dara, F.J.: Modelling groundwater level fluctuation in an indian coastal aquifer. Water SA **46**(4), 665–671 (2020). https://doi.org/https://doi.org/10.17159/wsa/2020.v46.i4.9081
12. Shwan Seeyan, Broder Merkel, R.A.: Investigation of the relationship betweengroundwater level fluctuation and vegetation cover by using ndvi for shaqlawa basin, kurdistan region – iraq. Geography and Geology **6**(3), 187–202 (2014). https://doi.org/10.5539/jgg.v6n3p187
13. Zhuoheng Chen, Stephen E. Grasby, K.G.O.: Relation between climate variabilityand groundwater levels in the upper carbonate aquifer, southern manitoba, canada. Hydrology **290**, 43–62 (2003). https://doi.org/10.1016/j.jhydrol.2003.11.02

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Análisis comparativo de alternativas para la detección y comunicación con dispositivos BLE Beacons en Aplicaciones Móviles Multiplataforma

Juan Fernández Sosa[1] [0000-0002-0482-3392], Santiago Medina[1] [0000-0001-6852-7165], Pablo Thomas[1] [0000-0001-9861-987X],
Leonardo Corbalán[1][0000-0001-9026-8059], Marcelo Naiouf[1] [0000-0001-9127-3212]

[1] Instituto de Investigación en Informática LIDI (III-LIDI). Facultad de Informática – Universidad Nacional de La Plata, La Plata, Argentina.
Centro Asociado a la Comisión de Investigaciones Científicas de la Provincia de Buenos Aires (CIC)
{jfernandez, smedina, pthomas, corbalan, mnaiouf}@lidi.info.unlp.edu.ar

Resumen. Los BLE beacons son pequeños dispositivos que utilizan la tecnología Bluetooth de bajo consumo para establecer comunicación con otros dispositivos compatibles. Para este fin, existen diferentes protocolos de comunicación, entre los cuales se destacan el protocolo iBeacon y Eddystone. La mayoría de los smartphones actuales incorporan tecnología Bluetooth de bajo consumo y pueden aprovechar aplicaciones de software específicas para detectar beacons y, de esta manera, ofrecer servicios basados en la proximidad o posicionamiento indoor, entre otras posibilidades. Este trabajo tiene como objetivo estudiar y analizar diferentes alternativas de interacción entre dispositivos móviles y los BLE beacons mediante una aplicación móvil multiplataforma desarrollada con un enfoque híbrido.

Keywords: BLE beacons, análisis comparativo, aplicaciones multiplataforma, servicios basados en la proximidad, entornos inteligentes, aplicaciones híbridas, iBeacon, Eddystone.

1. Introducción

El desarrollo de las tecnologías digitales en los últimos tiempos ha permitido avanzar sobre la digitalización e interconexión de cosas, espacios y experiencias. Un ejemplo de esto es la implementación de los servicios basados en la proximidad (PBS, por sus siglas en inglés), cuyo objetivo es alertar sobre la proximidad de un objeto o persona a un punto de interés. Este tipo de servicio permite crear soluciones que transforman los espacios en entornos inteligentes e interactivos [1]. Por ejemplo en museos [2] o centros de actividades tecnológicas [3], los visitantes pueden llevar consigo una aplicación en sus dispositivos móviles, y conforme se acercan a estos puntos de interés, reciben notificaciones e información relevante. Estas aplicaciones mejoran significativamente la experiencia de los visitantes al enriquecer su interacción con el entorno.

La implementación de este tipo de soluciones puede llevarse a cabo mediante pequeños dispositivos con capacidad de transmitir una señal utilizando tecnología Bluetooth de bajo consumo (BLE). Estos dispositivos son conocidos comúnmente como BLE Beacons y presentan diferentes protocolos de comunicación, entre los cuales

se destacan iBeacon (desarrollado por Apple) y Eddystone (desarrollado por Google). El funcionamiento de los beacons es sencillo: emiten periódicamente una señal con una pequeña cantidad de información que puede ser captada por otros dispositivos compatibles [4]. Los smartphones poseen la capacidad de interactuar con los beacons a partir de una aplicación de software (app) desarrollada específicamente para ese fin.

En el desarrollo de aplicaciones de software para dispositivos móviles existen dos grandes estrategias: el desarrollo nativo, y el desarrollo multiplataforma [5]. El primero se centra en la utilización de los entornos y herramientas de desarrollo que proveé cada uno de los sistemas operativos móviles especializados, siendo los más populares Android e iOS. Esto garantiza el acceso a todas las características del dispositivo, aunque se debe escribir un código fuente específico por cada una de éstas plataformas.

El desarrollo multiplataforma se contrapone al nativo y se centra en la reutilización de código. Dentro del desarrollo multiplataforma existen diferentes enfoques, uno de ellos es el enfoque híbrido basado en tecnologías web (HTML, Javascript y CSS). Las aplicaciones desarrolladas con este enfoque son ejecutadas en un contenedor web especial que brinda un mayor acceso a las capacidades del dispositivo (acelerómetro, bluetooth, cámara, GPS, entre otras) a través de plugins externos [6]. Por lo tanto, surge el interrogante de determinar si es posible desarrollar una aplicación móvil con un enfoque híbrido que permita detectar dispositivos BLE beacons y ofrecer, por ejemplo, servicios basados en la proximidad y posicionamiento indoor.

El objetivo de este trabajo consiste en realizar un análisis de posibilidades de implementación de una aplicación móvil que dé soporte a la detección de dispositivos BLE beacons empleando un enfoque de desarrollo multiplataforma del tipo híbrido.

El resto de este trabajo se organiza de la siguiente manera: en la sección 2 se presenta la tecnología Bluetooth de bajo consumo y los BLE beacons junto a sus principales protocolos de comunicación. Luego, en la sección 3, se describen los experimentos realizados y en la sección 4 el análisis de resultados de cada uno de ellos. Finalmente se mencionan conclusiones y trabajo futuro.

2. BLE beacons

Bluetooth es una tecnología de comunicación inalámbrica que fue presentada en 2010 y que se incluye dentro del estándar IEEE 802.15 que define las *Wireless Personal Area Network* (WPAN), redes orientadas a cubrir distancias cortas. Particularmente en IEEE 802.15.1 se especifica por completo la arquitectura y el funcionamiento de Bluetooth y Bluetooth de bajo consumo (BLE). Este último define en su capa física el uso de 40 canales de radiofrecuencia que operan en la banda libre ISM de 2.4GHz dedicada a industria, ciencia y medicina [7].

Una gran variedad de tipos de dispositivos como celulares, pulseras inteligentes o notebooks son compatibles con la tecnología BLE para la transmisión y recepción de datos. Además, esta tecnología es soportada por los sistemas operativos más utilizados en estos dispositivos, como por ejemplo Windows, Linux, Android e iOS.

Los beacons son pequeños dispositivos inalámbricos que se basan en el uso de la tecnología BLE para comunicación. Su pequeño tamaño, su bajo costo y la posibilidad

de alimentarlos con baterías por un largo tiempo vuelven su uso popular en aplicaciones que requieren de un mínimo caudal de datos [1].

El beacon emite un paquete cada cierto intervalo de tiempo con su identificador y en algunos casos también agrega datos telemétricos recogidos por sensores integrados por el fabricante. En los dispositivos receptores se ejecuta una aplicación de software para detectar la señal, recibir el paquete con el identificador y en función de su procesamiento, realizar una acción determinada.

La posibilidad de generar nodos sensores de bajo consumo energético que se pueden comunicar con una gran cantidad dispositivos heterogéneos llevó a BLE a convertirse en una de las tecnologías más establecidas en el mercado [8]. Estos nodos sensores se utilizan principalmente en el dominio de Internet de las Cosas (IoT) y Ciudades Inteligentes, ofreciendo numerosas posibilidades de aplicación, como el desarrollo de soluciones de marketing y publicidad, servicios basados en la proximidad, localización indoor, así como el posicionamiento y seguimiento de objetos, entre otras utilidades relevantes [7] [9] [10].

Para lograr una interacción fluida y segura entre los BLE beacons y otros dispositivos compatibles, existen diferentes protocolos de comunicación. Entre los cuales se destacan iBeacon y Eddystone [11]. Dichos protocolos se describen a continuación.

2.1 Protocolo iBeacon

iBeacon es un framework basado en BLE desarrollado por Apple y lanzado junto con el sistema operativo iOS 7 en 2013 [12].

El objetivo de iBeacon es ofrecer servicios basados en la proximidad para las aplicaciones móviles. Se utiliza la potencia de transmisión y el indicador de intensidad de señal recibida (RSSI), para determinar la distancia entre el iBeacon y el dispositivo móvil [13].

En la Figura 1 se muestra un detalle de la estructura del paquete de datos que utiliza el protocolo iBeacon.

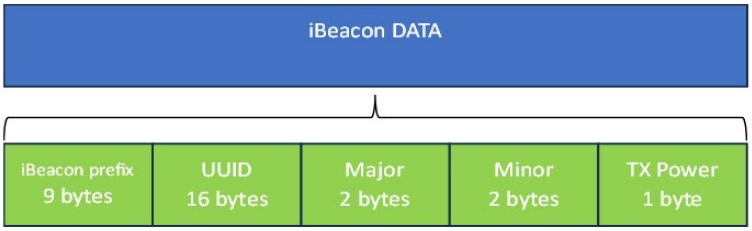


Fig 1. Paquete iBeacon

Los valores UUID, Major y Minor brindan información que permite a las aplicaciones identificar al iBeacon. Generalmente, esta información es de naturaleza jerárquica, los campos major y minor permiten subdividir la identidad establecida por el UUID de un beacon dentro de una aplicación específica. El TX Power es un parámetro que se utiliza para determinar la distancia del usuario al beacon. El prefijo

de iBeacon contiene flags, el header, el identificador del fabricante, el tipo de iBeacon y la longitud restante del paquete.

2.2 Protocolos Eddystone

Eddystone es un protocolo de BLE Beacon abierto lanzado por google en julio de 2015 compatible con Android e iOS [14].

Entre los objetivos de este protocolo se encuentran la implementación sencilla para una amplia variedad de dispositivos BLE, una arquitectura flexible que permite el desarrollo de nuevos tipos de mensajes y compatibilidad con la especificación de Bluetooth Core.

Define cuatro tipos de tramas distintas para diferentes aplicaciones: UID, URL, TLM y EID. Eddystone-TLM es la trama utilizada para transmitir información telemétrica sobre un beacon, como la temperatura, el voltaje y un conteo de los paquetes emitidos [15]. En la Figura 2 se puede ver un ejemplo de estructura de un frame TLM-Eddystone.

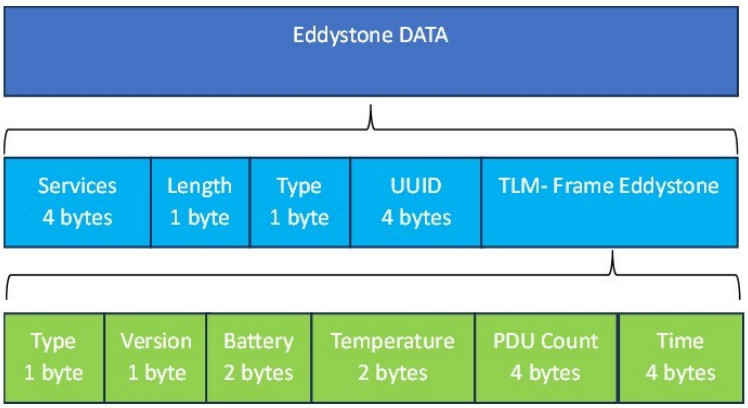


Fig. 2. TLM-Eddystone

3. Experimentación

3.1 Motivación

En el desarrollo de aplicaciones de software para dispositivos móviles, existen dos estrategias principales: la nativa y la multiplataforma [16] [17]. La estrategia nativa implica crear una aplicación específica para cada plataforma (sistema operativo) en la que se pretende distribuir el software. Por otro lado, la estrategia multiplataforma consiste en desarrollar un solo proyecto utilizando un mismo código fuente, el cual se ejecutará en diversas plataformas.

Mientras que en el enfoque nativo se emplean las herramientas de desarrollo (SDK) proporcionadas por cada plataforma, el enfoque multiplataforma ofrece mayor flexibilidad al permitir la utilización de diversas tecnologías. Específicamente, existe

una subcategoría dentro del desarrollo multiplataforma conocida como desarrollo de aplicaciones móviles del tipo híbridas que emplea tecnologías web estándar (HTML, JavaScript y CSS).

Las aplicaciones híbridas permiten la reutilización de código en las distintas plataformas, el acceso al hardware del dispositivo, y la distribución a través de las tiendas de aplicaciones. Sin embargo, para acceder al hardware del dispositivo en este tipo de desarrollo se deben utilizar plugins o librerías externas. A pesar de esta dependencia, el desarrollo híbrido ofrece la ventaja de requerir menos líneas de código para interactuar con el hardware del dispositivo en comparación con el enfoque nativo [18].

El objetivo de la experimentación en este trabajo consiste en identificar cuáles son las librerías o plugins que existen y permiten llevar a cabo la detección de dispositivos BLE beacons. Así mismo, se busca determinar cuáles de éstos podrían utilizarse para crear aplicaciones orientadas a dar soporte a servicios basados en la proximidad.

3.2 Tecnología seleccionada

Para el desarrollo de la aplicación se optó por utilizar el framework de código abierto Ionic [19]. Ionic es un framework que se emplea para desarrollar aplicaciones móviles híbridas mediante el uso de tecnologías web como HTML, CSS y

JavaScript. Esta herramienta proporciona un conjunto de componentes visuales que permiten lograr una apariencia similar a la de una aplicación nativa. Asimismo, ofrece la flexibilidad de integrar otros frameworks de desarrollo, entre los cuales se ha seleccionado Angular para este proyecto en particular.

Ionic se basa en Apache Cordova para acceder a las funcionalidades del hardware del dispositivo y permitir que las aplicaciones escritas en tecnologías web se ejecuten como aplicaciones móviles nativas. Cordova actúa como un puente entre el código web y las API nativas del dispositivo. Al compilar una aplicación con Cordova, se crea un contenedor nativo que embebe el código web y lo ejecuta como una aplicación móvil nativa en el dispositivo, esto permite que las aplicaciones desarrolladas con esta tecnología sean multiplataforma y se puedan ejecutar en diferentes sistemas operativos, utilizando de base el mismo código fuente.

3.3 Plugins evaluados

El proceso de búsqueda y selección de los plugins se llevó a cabo siguiendo un criterio específico. Se priorizó la elección de aquellos plugins que ofrecieran soporte para los protocolos de comunicación iBeacon y Eddystone, así como aquellos que permitieran la comunicación y detección de dispositivos utilizando la tecnología BLE. Se descartaron aquellos plugins que estaban diseñados exclusivamente para conectar con beacons de un fabricante específico, optando en su lugar por opciones más genéricas y versátiles.

Como resultado de este proceso se identificaron tres plugins que cumplían con los criterios planteados:

- **cordova-plugin-ibeacon**: plugin diseñado específicamente para la detección de dispositivos BLE beacons empleando el protocolo iBeacon [20].
- **cordova-plugin-eddystone**: plugin diseñado específicamente para la detección de dispositivos BLE beacons empleando el protocolo Eddystone [21]
- **cordova-plugin-ble-central**: este plugin permite la comunicación con dispositivos Bluetooth de bajo consumo, incluyendo Beacons. Proporciona funciones para escanear dispositivos BLE, conectarse a ellos y leer/escribir características [22].

Se realizó un análisis inicial de cada uno de estos plugins mediante la revisión de su documentación. Durante este proceso, se evaluaron diferentes datos, incluyendo la especificación de los protocolos de comunicación compatibles, así como su disponibilidad para las plataformas Android y/o iOS. Por otra parte, se recopiló el número de versión de cada plugin y la fecha de su última actualización. Estos detalles se sintetizan y presentan en la tabla 1.

Tabla 1. Listado de plugins seleccionados para la experimentación.

Plugin	**Versión**	**Última actualización**	**Protocolos compatibles**		**Plataformas compatibles**	
			iBeacon	**EddyStone**	**Android**	**iOS**
cordova-plugin-ibeacon	3.7.0	Julio 2018	Si	No	Si	Si
cordova-plugin-eddystone	1.3.0	Julio 2016	No	Si	Si	Si
cordova-plugin-ble-central	1.7.0	Mayo 2023	No indica	No indica	Si	Si

3.4 Diseño del experimento

La experimentación se centra en probar cada uno de los plugins seleccionados y verificar su capacidad para detectar un dispositivo BLE beacon, así como la información proporcionada por esta interacción. El objetivo es determinar la viabilidad para crear soluciones basadas en la proximidad utilizando dichos plugins.

Para alcanzar este objetivo, en primer lugar, fue necesario configurar y programar los dispositivos BLE beacons que se emplearían en los experimentos. Posteriormente, se procedió al desarrollo de la aplicación móvil que permitiera utilizar los plugins seleccionados. Ambos procesos son descritos en las subsecciones siguientes y el código fuente utilizado para construir tanto la aplicación móvil como los beacons se encuentra disponible en [23].

3.4.1 Configuración de BLE beacon

Para el despliegue de los BLE beacons se utilizó una placa de desarrollo basada en el microcotrolador ESP32 fabricado por Espressif.

El ESP32 combina la potencia de dos procesadores Xtensa de 32 bits con conectividad Wi-Fi y Bluetooth 4.0 integrada, lo que le permite comunicarse fácilmente con otros dispositivos y acceder a redes inalámbricas. Además, este microcontrolador se destaca por su amplia gama de interfaces y periféricos, que incluyen GPIOs, UART, I2C, SPI, ADC, DAC y más [24].

Para la experimentación se programaron dos beacons con el IDE Arduino. Cada uno de ellos utilizando un protocolo de comunicación específico: iBeacon y Eddystone-TLM. Como base para el desarrollo se utilizaron los ejemplos proporcionados por la librería BLE ESP32 para Arduino [25].

3.4.2 Desarrollo de la aplicación móvil

La aplicación móvil fue desarrollada utilizando la versión 6 del framework Ionic, y fue compilada para las plataformas Android e iOS. Se emplearon tres dispositivos móviles diferentes, cuyas características se mencionan en la tabla 2. En cada uno de estos dispositivos se evaluaron los tres plugins para la interacción con los beacons, resultando en un total de nueve pruebas realizadas.

Las pruebas sobre cada uno de los plugins corresponden con un número de experimento, que permitirá abordar, en la siguiente sección, los resultados: el experimento #1 y #2 utilizan los plugins *cordova-plugin-ibeacon* y *cordova-plugin-eddystone* respectivamente. El experimento #3 emplea el plugin para comunicación con dispositivos BLE genéricos *cordova-plugin-ble-central.*

Tabla 2. Listado de dispositivos móviles donde se realizaron los experimentos

Dispositivo	Sistema Operativo	Versión
iPhone 13 Mini	iOS	16.5.1
Moto G 9 Plus	Android	11
Huawei P9 Lite	Android	7

Los plugins fueron instalados siguiendo los pasos y recomendaciones detallados en la documentación oficial de cada uno de ellos. Incluyendo la gestión de los permisos necesarios para que funcionen correctamente.

La interfaz de la aplicación es sencilla y consta de tres botones, cada uno de los cuales habilita la detección de los beacons utilizando uno de los plugins seleccionados.

4. Resultados

A continuación, se presentan los resultados obtenidos luego de realizar las experimentaciones con cada uno de los plugins y los beacons configurados con los protocolos iBeacon y Eddystone.

4.1 Detección de BLE beacons en el experimento #1

En los resultados del experimento #1, se observó que la detección e identificación del beacon configurado con el protocolo iBeacon requería la especificación del valor del UUID, el valor mayor (Major) y el valor menor (Minor) dentro de la aplicación. Estos datos deben coincidir con la configuración establecida en el propio beacon para lograr una detección e identificación exitosa. Además, se encontró que es factible detectar varios beacons con un mismo UUID si solo se establece este parámetro.

La detección se logró realizar de manera exitosa tanto para los dispositivos con sistema operativo Android como para el dispositivo con iOS. Es decir que, para ambas plataformas, el plugin logró reconocer el beacon.

De la interacción entre la aplicación móvil y el beacon se obtuvieron un conjunto de datos en formato JSON, los cuales se presentan en la tabla 3. Si bien la detección se logró para ambas plataformas, se puede apreciar una diferencia en las respuestas para cada una de ellas. En el caso de Android se logró obtener los datos necesarios para determinar la distancia y la potencia de la señal (RSSI) del beacon. No obstante, en iOS sólo se logró la detección del beacon, sin recibir ningún dato que permita determinar la proximidad del dispositivo móvil al beacon.

Tabla 3. Resultado del experimento #1

Plugin	**Detección BLE Beacon Protocolo iBeacon**		**Resultado de la interacción**	
	Android	**iOS**	**Android**	**iOS**
cordova-plugin-ibeacon	Si	Si	{ **"uuid"**:"4d6fc88bbe75-5555-4444-3333a36ec78e", **"major"**:"0", **"minor"**:"53", **"proximity"**:"ProximityFar", **"rssi"**:-66, **"tx"**:0, **"accuracy"**:9223372036854776 0 }	{ **"minor"**:15, **"rssi"**:0, **"major"**:0, **"proximity"**:"ProximityUnknown", **"accuracy"**:-1, **"uuid"**:"4D6FC88B-BE75-5555-444 4-3333A36EC78E "}

4.2 Detección de BLE beacons en el experimento #2

A diferencia del experimento #1, para detectar los beacons mediante el protocolo Eddystone, no hace falta especificar el identificador del beacon, sino que se hace un escaneo genérico intentando identificar a los beacons presentes.

Sin embargo, durante las pruebas realizadas, se encontró que no fue posible detectar la presencia del beacon configurado con el protocolo Eddystone. El plugin utilizado reportó un error de compatibilidad con otras librerías, lo que impidió la realización de la detección. Es relevante mencionar que este plugin recibió su última actualización en

el año 2016, lo que sugiere una falta de mantenimiento y, por consiguiente, lo descarta como una opción viable para este propósito.

Tabla 4. Resultados del experimento #2

Plugin	**Detección BLE Beacon Protocolo Eddystone**		**Resultado de la interacción**	
	Android	**iOS**	**Android**	**iOS**
cordova-plugin-eddystone	No	No	Sin resultados	Sin resultados

4.3 Detección de BLE beacons en el experimento #3

En el tercer experimento, se utilizó un plugin que no fue específicamente diseñado para la detección de beacons, pero que ofrece la funcionalidad de detectar dispositivos que utilizan Bluetooth de bajo consumo. Este plugin fue probado para detectar tanto el beacon configurado con el protocolo iBeacon, como aquel configurado con el protocolo Eddystone. Los resultados de la detección, tanto para Android como para iOS, se muestran en la tabla 5, y en las tablas 6a y 6b se presentan los datos obtenidos durante la interacción.

El plugin permitió realizar un escaneo general de todos los dispositivos cercanos al dispositivo móvil donde se ejecutó la aplicación. Por esta razón, muchos de los resultados detectados pertenecían a dispositivos no buscados. No obstante, al aplicar filtros adecuados, fue posible reconocer y aislar los beacons de interés.

En el caso de los dispositivos con Android, el plugin logró detectar a los beacons configurados con ambos protocolos de comunicación. La identificación de los beacons se llevó a cabo mediante el ID del beacon que corresponde a la dirección MAC del dispositivo. En el caso de los beacons Eddystone, la respuesta incluye además un atributo con el nombre del dispositivo, que es configurable desde el lado del beacon, lo que facilitó su identificación.

Con el dispositivo iOS los resultados al utilizar el plugin presentaron algunas diferencias. El plugin permitió detectar correctamente al beacon configurado con el protocolo Eddystone. Al igual que en las pruebas realizadas para Android, la identificación y filtrado de los resultados se llevaron a cabo utilizando el nombre configurado para el beacon.

En cuanto a los iBeacons, durante los experimentos en iOS, inicialmente se enfrentó a un desafío para identificar cuál de los dispositivos detectados correspondía al beacon configurado. Esto se debió a que el valor del ID que se retornaba no era el UUID real del dispositivo, sino uno generado adrede por el sistema operativo. Dicho valor puede variar en diferentes dispositivos móviles que estén intentando detectar al beacon, por lo que el proceso de detección fue dificultoso. Para resolver este problema, se realizó una modificación en el beacon configurado con el protocolo iBeacon para que devuelva un adicional en la interacción, el nombre del beacon. Con esta modificación del lado del beacon fue posible identificarlo de manera certera en la aplicación.

Tanto para dispositivos Android como para iOS, una vez que se aislaron los resultados de los beacons, se pudo observar que ambos protocolos devuelven la

potencia de la señal detectada (RSSI). Este dato es valioso, ya que permite estimar la distancia entre el dispositivo móvil y el beacon, y resulta crucial para diversas aplicaciones que se basan en la proximidad del usuario a los beacons detectados.

Tabla 5a. Resultados del experimento #3

Plugin	**Detección BLE Beacon Protocolo iBeacon**		**Detección BLE Beacon Protocolo Eddystone**	
	Android	**iOS**	**Android**	**iOS**
cordova-plugin-ble-central	Si	Si*	Si	Si

Tabla 6a. Resultados del experimento #3 - iBeacon

Plugin	**Resultados interacción iBeacon**	
	Android	**iOS**

cordova-plugin-ble-central	{
	"**id**":"30:C6:F7:21:23:F2",
	"**advertising**":{},
	"**rssi**":-27
	}
		{
		advertising = {
		kCBAdvDataIsConnectable = 0;
		kCBAdvDataLocalName = "ESP32-iBeacon";
		};
		id = "3FFB9E7F-31AD-F11A-85B1-E4039055AF28";
		name = "ESP32-iBeacon";
		rssi = "-69";
		state = disconnected;
		}

Tabla 6b. Resultados del experimento #3 - Eddystone

Plugin	**Resultados interacción Eddystone**	
	Android	**iOS**

cordova-plugin-ble-central	{
	"**name**":"ESP32",
	"**id**":"30:C6:F7:21:3C:46",
	"**advertising**":{},
	"**rssi**":-74
	}
		{
		advertising = {
		kCBAdvDataIsConnectable = 1;
		kCBAdvDataLocalName = ESP32;
		};
		id = "3FFB9E7F-31AD-F11A-85B1-E4039055AF28";
		name = ESP32;
		rssi = "-69";
		state = disconnected;
		}

4.4 Análisis general de resultados

El análisis de los resultados obtenidos en cada uno de los experimentos permite formular las siguientes recomendaciones: en el caso de desarrollar una aplicación para la detección de dispositivos BLE beacons utilizando el protocolo Eddystone, se aconseja no emplear el plugin "*cordova-plugin-eddystone*". Sin embargo, como alternativa viable, se sugiere utilizar el plugin de detección general de beacons "*cordova-plugin-ble-central"*, y aplicar un proceso de filtrado según el nombre del beacon. Esta recomendación se extiende tanto para plataformas Android como para iOS.

En el contexto de los iBeacons, para aplicaciones orientadas a la plataforma Android, se recomienda el uso del plugin específico para este tipo de beacons, "*cordova-plugin-ibeacon*". Este plugin permite la detección de iBeacons y, además, incluye datos en la respuesta que facilitan el cálculo de la distancia entre el dispositivo y el beacon. Por ende, es posible considerar este plugin para la creación de soluciones que aprovechen la proximidad para brindar servicios basados en esta característica. No obstante, para dispositivos iOS, este plugin solo posibilita la detección del beacon, sin proporcionar una forma precisa de determinar la distancia entre el dispositivo móvil y el mismo.

Para superar esta limitante en iOS, se recomienda emplear el plugin "*cordova-plugin-ble-central*", el cual, según los resultados de las pruebas realizadas, demostró la capacidad de identificar tanto beacons con protocolo iBeacon como Eddystone. En el caso específico de los iBeacons en iOS, es fundamental configurar el beacon para que muestre el dato del nombre, de modo que pueda ser identificado de manera certera por la aplicación.

A pesar de que el plugin utilizado no fue específicamente diseñado para la detección de beacons, su capacidad para detectar dispositivos Bluetooth de bajo consumo y reconocer beacons configurados con diferentes protocolos resultó ser una ventaja significativa en el contexto del estudio. No obstante, es fundamental tener en cuenta las limitaciones inherentes a un escaneo general, y la necesidad de aplicar filtros adecuados para obtener resultados precisos y relevantes en la detección e identificación de los beacons de interés.

5. Conclusiones y Trabajo Futuro

En este trabajo se presentó un análisis de diferentes opciones que existen para desarrollar una aplicación de software para dispositivos móviles que sea capaz de detectar dispositivos BLE beacons. Particularmente el análisis se focalizó en el desarrollo de una aplicación multiplataforma del tipo híbrido.

Los BLE beacons poseen diferentes protocolos que permiten a las aplicaciones poder detectarlos e intercambiar información con ellos. En particular, en este trabajo se estudiaron los protocolos iBeacon y Eddystone.

La experimentación consistió en el análisis y prueba de tres plugins diferentes: a) *cordova-plugin-ibeacon, b) cordova-plugin-eddystone y c) cordova-plugin-ble-central.* Estos plugins fueron probados en tres dispositivos móviles diferentes, dos de ellos con

sistema operativo Android y uno con sistema operativo iOS. La aplicación desarrollada, fue compilada y ejecutada en ambos sistemas operativos.

Es necesario considerar que los resultados presentados en este trabajo, están ligados a las versiones tanto del framework Ionic como de los plugins utilizados al momento de llevar a cabo la experimentación y, por lo tanto, podrían variar en el futuro, conforme éstos reciban actualizaciones.

A partir de los resultados de las experimentaciones se puede establecer una serie de recomendaciones, en lo relativo al desarrollo de una aplicación móvil para la detección de BLE beacons empleando el framework Ionic. Si se tiene que interactuar con un beacon con protocolo Eddystone es conveniente utilizar el plugin *cordova-plugin-ble-central*, ya que se logró una detección del beacon tanto en los dispositivos Android como en el dispositivo iOS. Además, este plugin permite obtener los datos de la intensidad de la señal que permite calcular distancias para ofrecer servicios basados en la proximidad. Este plugin se utiliza como alternativa al plugin específico para la detección de beacons con el protocolo Eddystone *cordova-plugin-eddystone,* el cual tuvo que ser descartado al no poder ejecutar el plugin correctamente.

Por otra parte, en el caso de los iBeacons hay que diferenciar el comportamiento en dispositivos Android de iOS. En Android, para la detección de iBeacons se recomienda utilizar el plugin *cordova-plugin-ibeacon*, ya que con éste se logró detectar los beacons y obtener información relevante para determinar la distancia del dispositivo móvil al beacon. En iOS si bien la detección de los iBeacons se logró con éxito, no así la obtención de los datos necesarios para calcular la proximidad al beacon. Para superar esta limitante en iOS, se recomienda emplear el plugin *cordova-plugin-ble-central* en caso de poder manipular la configuración del beacon agregando un nombre para poder identificar a este dispositivo de los otros que este plugin puede detectar.

Como trabajo futuro se plantea continuar con el análisis de librerías o plugins para otros Frameworks de desarrollo de aplicaciones móviles multiplataforma. También se pretende desarrollar diferentes tipos de beacons basados en otros microcontroladores que están disponibles en el mercado.

Referencias

[1] Spachos, P., & Plataniotis, K. N. (2020). BLE beacons for indoor positioning at an interactive IoT-based smart museum. IEEE Systems Journal, 14(3), 3483-3493.

[2] Steiniger, S., Neun, M., & Edwardes, A. (2006). Foundations of location based services. Lecture Notes on LBS, 1(272), 2.

[3] Fernández Sosa, J. F., Medina, S., Segovia, M., Thomas, P. J., & De Giusti, A. E. (2023). Propuesta e implementación de un sistema basado en servicios de proximidad con BLE Beacons. In XXVIII Congreso Argentino de Ciencias de la Computación (CACIC)(La Rioja, 3 al 6 de octubre de 2022).

[4] Lindh, J. (2015). Bluetooth low energy beacons. Texas Instruments, 2.

[5] Thomas, P. J., Delía, L. N., Corbalán, L. C., Cáseres, G., Fernández Sosa, J., Tesone, F., ... & Pesado, P. M. (2018). Tendencias en el desarrollo de Aplicaciones para Dispositivos

Móviles. In XX Workshop de Investigadores en Ciencias de la Computación (WICC 2018, Universidad Nacional del Nordeste)..

[6] Kaczmarczyk, A., Zając, P., & Zabierowski, W. (2022). Performance Comparison of Native and Hybrid Android Mobile Applications Based on Sensor Data-Driven Applications Based on Bluetooth Low Energy (BLE) and Wi-Fi Communication Architecture. Energies, 15(13), 4574.

[7] Darroudi, S. M., & Gomez, C. (2017). Bluetooth low energy mesh networks: A survey. Sensors, 17(7), 1467.

[8] Aguilar, S., Vidal, R., & Gomez, C. (2017). Opportunistic sensor data collection with bluetooth low energy. Sensors, 17(1), 159.

[9] Chang, K. H. (2014). Bluetooth: a viable solution for IoT?[Industry Perspectives]. IEEE Wireless Communications, 21(6), 6-7.

[10] Akinsiku, A., & Jadav, D. (2016, April). BeaSmart: A beacon enabled smarter workplace. In NOMS 2016-2016 IEEE/IFIP Network Operations and Management Symposium (pp. 1269-1272). IEEE.

[11] Jeon, K. E., She, J., Soonsawad, P., & Ng, P. C. (2018). Ble beacons for internet of things applications: Survey, challenges, and opportunities. IEEE Internet of Things Journal, 5(2), 811-828. https://developer.apple.com/ibeacon/Getting-Started-with-iBeacon.pdf

[12] Kwok, C. Y. T., Wong, M. S., Griffiths, S., Wong, F. Y. Y., Kam, R., Chin, D. C., ... & Mok, E. (2020). Performance evaluation of iBeacon deployment for location-based services in physical learning spaces. Applied Sciences, 10(20), 7126. [14] Hernández-Rojas, D. L., Fernández-Caramés, T. M., Fraga-Lamas, P., & Escudero, C. J. (2017). Design and practical evaluation of a family of lightweight protocols for heterogeneous sensing through BLE beacons in IoT telemetry applications. Sensors, 18(1), 57. https://github.com/google/eddystone/blob/master/protocol-specification.md

[13] Delia, L., Galdamez, N., Thomas, P., Corbalan, L., & Pesado, P. (2015, Mayo).

[14] Multi-platform mobile application development analysis. In 2015 IEEE 9th International Conference on Research Challenges in Information Science (RCIS) (pp. 181-186). IEEE.

[15] Delia, L., Thomas, P., Corbalan, L., Sosa, J. F., Cuitiño, A., Cáseres, G., & Pesado, P. (2018, Julio). Development approaches for mobile applications: comparative analysis of features. In Science and Information Conference (pp. 470-484). Springer, Cham.

[16] Kaczmarczyk, A., Zając, P., & Zabierowski, W. (2022). Performance Comparison of Native and Hybrid Android Mobile Applications Based on Sensor Data-Driven Applications Based on Bluetooth Low Energy (BLE) and Wi-Fi Communication Architecture. Energies, 15(13), 4574.

[17] https://ionicframework.com/, accedido en Julio 2023.

[18] https://github.com/petermetz/cordova-plugin-ibeacon, accedido en Julio 2023.

[19] https://github.com/evothings/cordova-eddystone, accedido en Julio 2023.

[20] https://github.com/don/cordova-plugin-ble-central, accedido en Julio 2023.

[21] https://gitlab.com/iii-lidi/papers/beacons-app-cacic2023

[22] https://espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf, accedido en Julio 2023.

[23] https://github.com/nkolban/ESP32_BLE_Arduino, accedido en Julio 2023.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

An empirical study of API Management and ISO/IEC SQuaRE: a practitioners' perspective

Eder dos Santos[1,2] and Sandra Casas[1]

[1] Instituto de Tecnología Aplicada - Universidad Nacional de la Patagonia Austral
http://www.uarg.unpa.edu.ar/ita

[2] Centro de Investigaciones Científicas y Técnicas(CIT) Santa Cruz - Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET) Río Gallegos, Argentina
{esantos,sicasas}@uarg.unpa.edu.ar

Abstract. APIs have become a cornerstone in software ecosystems: organizations have been increasingly connecting software applications to share complex digital assets. However, managing APIs is non-trivial and companies have been struggling with different quality-related issues in managing their APIs. Therefore, our goal aims to understand how practitioners perceive quality characteristics related to API management functions. To achieve it, we adopted the ISO/IEC 25010 (SquaRE) standard as the reference model and conducted a structured survey with professional developers, system administrators and software functional analysts based in Rio Gallegos City, Argentina. Questions were based on main API management capabilities discussed in current literature and their relation with the ISO/IEC 25010 quality characteristics. After running an anova analysis over 136 answered topics, results show that Functional Suitability and Security were perceived as the most critical quality capabilities, which can provide a basis for future research.

Keywords: API Management, Software Quality, SQuaRE

1. Introduction

In the last years, distribution models of information systems are evolving into XaaS [1] paradigms, in which organizations make their digital assets available to customers [2]. In this context, the microservices architectures are rapidly emerging as they provide a flexible architectural model [3] which allows the distribution of an organization's information systems into a highly-scalable set of services. In general, such services are supported by APIs that typically stand by REST [4] principles.

According to [5], APIs present both a technical side and a business side. The first one refers to "a technical answer to a business problem" and can be understood as a set of requirements that govern how applications can interact and exchange data, while the latter can be considered "a business enabler" as it determines how the organizations want to use their assets to deliver value not only across internal organizational units but also to external third parties. In this sense, the term API Economy is being used to describe the increasing movement of the industries towards new ways of expanding their business models by conceiving and commercializing solutions that expose their APIs as part of their service offering [6].

The API Economy scenario exerts additional pressure on the development, deployment and maintenance of information systems. Since APIs have gained a critical

aspect [7], organizations need to mitigate the risks of failure by managing their APIs through API management [5][8] platforms. API Management Platforms provide the basic capabilities to create, analyze, and manage APIs in a secure and scalable environment such as providing helpful documentation, controlling access to the API, as well as monitoring and analyzing its usage [8][9][10].

Notwithstanding the above, API management activities imply a series of challenges in both internal [5][11] and descentralized [12] software ecosystems. In this sense, software quality models comprise a set of characteristics and sub characteristics that makes up a platform on which quality assessments about software components can be performed, as a critical step in providing pragmatic solutions that can be readily adopted by software practitioners in industry. Quality models also allow for continuous monitoring of the quality of an organization's software components, which facilitates quick intervention when deviations are encountered. Therefore, it is crucial to inquire how practitioners in the software industry perceive quality characteristics related to common functions and capabilities of API management platforms, in order to help defining benchmarks and to provide insight into emerging trends at various stages of the software product's lifecycle.

In this direction, this paper shows the results of the first survey focused on a group of professionals based in the Santa Cruz province, Argentina. It is an initial exploratory instance aimed, in particular, at identifying the relationship between the quality characteristics established in the quality model described in the ISO/IEC 25010 standard and the API management main functions / capabilities.

This paper is structured as follows: Section 2 describes the methodology employed in designing and conducting the survey aimed at establishing itself as a preliminary study, which will serve as the basis for more specific works in the future. Results are detailed in section 3, and section 4 presents a brief discussion about API surveys, analysis of results, threats to validity, limitations, thoughts about the impact and significance of this work, and concluding remarks and future directions.

2. Materials and Methods

This section describes the methodology employed in conducting a survey aimed at establishing itself as a preliminary study, which will serve as the basis for more specific works in the future. The survey follows an exploratory approach based on the classification proposed by [14]. The purpose of the survey is to gather insights into the API management practices and assess the quality characteristics of API management platforms. The results obtained from this preliminary study will serve as a foundation for future research endeavors that delve deeper into specific aspects of API management.

Sampling Technique: The survey utilized a non-probabilistic sampling scheme, specifically convenience sampling, as suggested by [14][15][16]. Convenience sampling involves selecting participants based on their accessibility. In this study, subjects were selected from professionals based in the province of Santa Cruz, Argentina, who have an academic background in computer science and currently work

in the area, both in the public and private spheres. This specific sample was chosen due to their expertise and relevance to the study's focus.

Data Collection: The survey implementation and data collection were conducted using an online survey platform, ensuring ease of access and facilitating data capture.

Survey Design: To compile the survey questions, the study incorporated the eight main quality characteristics described in ISO/IEC 25010 and a set of 15 capabilities supported by API management platforms as outlined in the literature. The survey consisted of two main groups of prompts, detailed as follows.

The first group of prompts comprised five open-ended questions aimed at collecting demographic data and gathering information about the professional trajectories of the respondents. The purpose of these questions was to obtain comprehensive census data and gain insights into the respondents' backgrounds, focusing on their experiences within the API management domain.

The second group consisted of 15 closed-ended multiple-choice grid questions, with each question corresponding to one of the main API management capabilities [8][9][13]. Selected capabilities were: 1. Authentication; 2. Authorization (Access Control); 3. Identity mediation; 4. Data privacy through encryption; 5. Data privacy through masking; 6. Key and certificate management; 7. DoS protection; 8. Threat detection; 9. Consumption quota; 10. Spike arrest; 11. Usage throttling; 12. Traffic prioritization; 13. Format translation; 14. Protocol translation; and 15. Service and data mapping.

Each grid contained eight lines with Yes/No (Related / Not related) options, representing the ISO/IEC 25010 quality characteristics. This design allowed for the assessment of the extent to which each API management function aligned with each quality characteristic, and mandatory radio buttons were used. A sample question can be seen in table 1.

In total, each respondent should answer 125 questions, including both openended and closed-ended questions. The survey was made available for a period of two weeks to ensure a sufficient response rate and gather a representative sample. This duration allowed potential respondents, who are professionals based in the province of Santa Cruz, Argentina, with academic training in computer science and working in the area, both in the public and private spheres, ample time to access the survey and complete the questionnaire at their convenience.

1. Authentication: Authentication is the process of uniquely determining and validating the		
ISO Characteristic	Related	Not related
1. Functional Suitability		
2. Performance Efficiency		
3. Compatibility		
4. Usability		
5. Reliability		
6. Security		
7. Maintainability		
8. Portability		

Table 1. A survey's question sample format.

3. Results

The survey was conducted as a pilot experiment and involved a small sample of professionals working in the IT-related industry. A total of six participants took part in the study. The demographic and professional characteristics of the participants are summarized as follows:

Age: The age range of the participants varied from 27 to 42, with a mean age of 31 years.

Experience: The professional experience of the participants ranged from 4 to 20, with an average experience of 9 years.

In relation with labor sphere, It is important to note that some participants provided multiple answers for both the labor sphere and role questions, indicating their involvement in various domains or responsibilities within the IT-related industry. Results are detailed as follows:

Private Companies (28.57 %), Government Departments: (71.43 %).

Role (self-described): Developer (coding, full stack: 50.00 %), Help and Support: (25.00 %), Functional Analyst: (8.33 %), Project Management: (8.33 %), Research and Development (8.33 %).

Findings: As this study was a pilot experiment aimed at gathering preliminary insights, the findings emerged from the data collected can be seen in following figures 1 and 2. Figure 1 displays the survey responses for the "Authentication" capability. The responses are presented as a count (% of total responses), and a corresponding 3D bar graph is provided. The horizontal axis of the graph represents the quality characteristics described in the ISO 25010 standard, while the bars represent the responses. Additionally, error bars are incorporated into the graph.

Due to the extensive nature of the survey, which covered 15 capabilities, the data from all positive responses has been synthesized and presented in Figure 2. To enhance the visualization and understanding of the results, a color scale similar to that used in Figure 1 was adopted, with the midpoint set at the 20th percentile. The figure includes the average and median values for reference.

1. Authentication		
ISO 25010 Characteristic	Related	Not Related
FS. Functional Suitability	100,00%	0,00%
RE. Reliability	33,33%	66,67%
PE. Performance Efficiency	50,00%	50,00%
US. Usability	0,00%	100,00%
SE. Security	83,33%	16,67%
CO. Compatibility	16,67%	83,33%
MA. Maintainability	33,33%	66,67%
PO. Portability	0,00%	100,00%



Fig.1. Survey results for capability 1 - Authentication - and its relation with ISO 25010 quality characteristics.

4. Discussion

The utilization of surveys directed towards API practitioners is a commonly employed initiative by industry-leading organizations. These surveys result in influential guiding documents for both the web API production and consumption sectors. In our study, we have incorporated select items from the RapidAPI survey to enrich our analysis.

On a global scale, "The Rapid Developer Survey" [23] garnered 850 responses from over 100 different countries. A substantial majority of respondents identified as professional developers, with over 85% reporting active programming as part of their vocation or educational pursuits. The survey participants represented a diverse spectrum of organizational sizes, experience levels, and industries. Furthermore, the company Smartbear annually publishes their report titled "State of Software Qualit - API" [24] with the goal of "Identifying the latest benchmarks and gaining insights into the trajectory of the API industry." This report collates information on "the latest methodologies, practices, and tools used by software teams across the world." The survey responses encompassed more than 1,100 API practitioners and customers, spanning various industries, and ranged from startups to large enterprises in terms of company size. Finally, within an argentinian perspective, [25] presented the outcomes of the first-ever survey specifically targeting developers in Argentina. The primary objective of this investigation was to comprehend the usage patterns and challenges faced by software developers when consuming web APIs within the Argentinean context.

Despite the existence of prior surveys, there is still a dearth of evidence pertaining to investigations specifically targeting the perceived quality of stakeholders in the industry, as proposed in this current work and our ongoing research agenda. Our focus is on characterizing quality aspects concerning API management requirements, capabilities, and best practices, utilizing the ISO 25010 / SQuaRE standard as our quality reference model.

It is important to note that these findings should be interpreted with caution due to the small sample size and the exploratory nature of the pilot experiment.

		ISO 25010 Characteristics							
		Functional Suitability	Reliability	Performance Efficiency	Usability	Security	Compatibility	Maintainability	Portability
API Management Capabilities	1. Authentication	100,00%	33,00%	50,00%	0,00%	83,00%	17,00%	33,00%	0,00%
	2. Authorization	100,00%	33,00%	67,00%	33,00%	100,00%	17,00%	17,00%	0,00%
	3. Identity mediation	100,00%	17,00%	50,00%	17,00%	100,00%	67,00%	83,00%	17,00%
	4. Data encryption	100,00%	17,00%	67,00%	50,00%	83,00%	17,00%	17,00%	0,00%
	5. Data masking	100,00%	50,00%	17,00%	33,00%	100,00%	17,00%	0,00%	0,00%
	6. Keys and certificates	100,00%	83,00%	0,00%	67,00%	100,00%	17,00%	17,00%	17,00%
	7. DoS protection	100,00%	50,00%	83,00%	0,00%	100,00%	17,00%	17,00%	17,00%
	8. Threat detection	100,00%	50,00%	50,00%	0,00%	100,00%	17,00%	17,00%	0,00%
	9. Consumption quota	50,00%	50,00%	100,00%	67,00%	83,00%	17,00%	0,00%	0,00%
	10. Spike arrest	50,00%	67,00%	100,00%	0,00%	83,00%	17,00%	0,00%	0,00%
	11. Usage throttling	83,00%	50,00%	100,00%	17,00%	100,00%	67,00%	0,00%	0,00%
	12. Traffic prioritization	100,00%	67,00%	100,00%	33,00%	33,00%	0,00%	0,00%	0,00%
	13. Format translation	100,00%	17,00%	0,00%	83,00%	0,00%	100,00%	50,00%	33,00%
	14. Protocol translation	100,00%	17,00%	100,00%	50,00%	0,00%	100,00%	50,00%	17,00%
	15. Service and data mapping	83,00%	0,00%	0,00%	100,00%	0,00%	67,00%	17,00%	0,00%
	Average	91,07%	40,07%	58,93%	36,67%	71,00%	36,93%	21,20%	6,73%
	Mean	100,00%	50,00%	67,00%	33,00%	83,00%	17,00%	17,00%	0,00%

Fig.2. Positive results API Management capabilities and their relation with ISO 25010 quality characteristics.

They provide initial insights that can guide further research in this area. In this study, our primary objective was to identify which quality characteristics were perceived as related to API Management capabilities, according to the opinions of the participants. The evaluation encompassed 15 API management capabilities, and participants were asked to indicate their relevance to the eight ISO 25010 quality characteristics.

The quality characteristic of Functional Suitability received the highest average percentage of related responses at 91.07%. This suggests that a significant portion of the evaluated API Management capabilities align well with meeting the intended functionality and user needs. The mean value of 100% further supports the strong association between the capabilities and the quality characteristic of functional suitability. We hypothesized that capabilities, serving as functional requirements of API Management software products, would exhibit a high rate of functional suitability. The results strongly support this hypothesis.

Security obtained a relatively high average percentage of related responses, at 71%. This indicates that a substantial number of the API Management capabilities included in this study contribute to ensuring security aspects within the platforms. The mean value of 83% further confirms the strong alignment with the security quality characteristic.

Reliability garnered an average percentage of related responses at 40.07%. Though lower than Functional Suitability, the results still indicate a moderate relationship between API Management capabilities and reliability. The mean value of 50% implies that half of the evaluated capabilities are considered relevant to ensuring the reliable performance of API management platforms.

Participants attributed an average percentage of 58.93% to Performance Efficiency, indicating that many API Management capabilities contribute to efficient performance. The mean value of 67% signifies that a significant portion of capabilities supports the quality characteristic of performance efficiency.

Usability received an average percentage of related responses at 36.67%. This suggests that some API Management capabilities are associated with the userfriendliness and ease of use of the platforms. The mean value of 33% indicates a moderate level of relevance between the capabilities and the usability quality characteristic.

Compatibility received an average percentage of related responses at 36.93%. This suggests that certain API Management capabilities contribute to compatibility with external systems and technologies. The mean value of 17% indicates a relatively lower level of relevance with the compatibility quality characteristic. Maintainability garnered an average percentage of related responses of 21.2%. This implies that some API Management capabilities relate to the ease of maintenance and updates. The mean value of 17% reinforces the moderate level of alignment with the maintainability quality characteristic.

Portability obtained the lowest average percentage of related responses, with only 6.73%. This suggests that very few API Management capabilities are seen as related to portability. The mean value of 0% confirms the limited alignment with the portability quality characteristic.

These findings provide valuable insights into the perceived associations between API Management capabilities and the ISO 25010 quality characteristics. It is essential to consider these results when designing, developing, and enhancing API management platforms to better meet user needs and ensure high-quality performance.

4.1 Threats to Validity and Limitations

When conducting surveys, it is important to consider the potential threats to the validity of the results. In this sense, the sample and population size can be particularly considered as potential threats to the validity of survey results. Specifically, two types of threats related to sample and population size are worth considering: a) Sampling Bias [17][18] can limit the generalizability of the findings and lead to inaccurate conclusions about the broader population. b) Limited Statistical Power [19] [20]: the study may lack sufficient statistical power to detect meaningful relationships or differences, which can increase the likelihood of Type II errors (false negatives) and limit the ability to draw accurate conclusions from the data.

To mitigate this threat, we have implemented various strategies as described in [21]. These strategies include utilizing personal networks, constructing a sample based on convenience, authority, and credibility. Given that our study is an academic research project conducted by a national university, we included all necessary research details in the questionnaire. Additionally, we designed a brief and concise survey, carefully considering the type and number of instructions provided. Moreover, it is worth noting that the sample size in our study is comparable to other surveys targeting software developers using APIs, as reported in the literature. For instance, in [22], the sample consisted of 6 participants.

Additionaly, to mitigate the risk of false negatives and eliminate errors due to ambiguous responses, the survey questions were designed as mandatory radio button selections. This design choice aimed to ensure clear and unambiguous choices between

"Related" and "Not Related" options, reducing the potential for misinterpretation or inaccurate responses.

While the study provides valuable contributions, it is not without limitations. Given the pilot nature of the experiment, it is essential to acknowledge the limitations inherent in the study design. These limitations include: a) Small sample size: The study involved only six participants, which may limit the generalizability of the findings. b) Lack of diversity: The sample consisted of a reduced group of participants, which may not be representative of the broader population of professionals in the IT-related industry. c) Preliminary nature: The study aimed to gather preliminary insights and may not provide definitive conclusions.

4.2 Impact and Significance

The findings of this study have significant implications for the field of API management and software development. Understanding the relationships between API Management capabilities and ISO 25010 quality characteristics provides valuable insights into the design, development, and enhancement of API management platforms as software products. The study's insights on the associations between capabilities and ISO 25010's quality characteristics can guide platform providers in making informed decisions to improve specific aspects of their API management offerings. By addressing the identified gaps and weaknesses, developers can work towards creating more robust, efficient, and user-friendly API management platforms that meet users' expectations and industry standards.

Additionally, the study's results can help API management platform users, developers, and decision-makers make informed choices when selecting or customizing platforms to align with their specific needs and quality requirements. The understanding of which capabilities contribute significantly to each quality characteristic empowers stakeholders to make data-driven decisions in their platform evaluation and adoption processes. In this sense, confirming the identified high rate of functional suitability and security among capabilities through wider surveys may indicate that developers and organizations can prioritize and invest in functionalities that align well with user needs and overall system functionality.

4.3 Concluding Remarks and Future Directions

In conclusion, this study successfully identified the quality characteristics that participants perceived as related to API Management capabilities. The high rate of functional suitability among capabilities confirms our hypothesis, emphasizing the platforms' strong alignment with meeting functional needs and user requirements.

The results of this study contribute to the existing body of knowledge in API management and software development. While the study has its limitations, such as the relatively small sample size, the findings lay the groundwork for future research in this domain. For instance, the high rate of security relatedness may establish new hypothesis in future work.

The moderate associations found between capabilities and other ISO 25010 quality characteristics present opportunities for improvement. Addressing the identified gaps

can lead to enhanced API Management functional suitability, reliability, performance efficiency, usability, security, compatibility, maintainability, and portability.

We recommend that future studies explore these associations with larger and more diverse samples to enhance the generalizability of the results. Additionally, considering other factors that may influence participants' perceptions, such as their experience with API management platforms, could provide further insights into quality characteristic prioritization.

In conclusion, the insights gained from this study offer valuable guidance to API management platform providers, developers, and users, promoting the continuous improvement and adoption of high-quality API management software products that meet the evolving needs of the software industry.

Based on the preliminary findings and the identified limitations, future research in this area should consider the following directions: a) Increase sample size: It is important to address the potential threats by carefully considering the sample size and ensuring that it is appropriate for the research objectives and the target population. Conducting the study with a larger and more diverse sample would enhance the generalizability of the results, as larger sample sizes can help improve the representativeness of the sample and enhance the statistical power of the study, reducing the risk of sampling bias and increasing the likelihood of detecting meaningful effects. b) Explore additional variables: Investigate other factors or variables that may influence the observed patterns to gain a more comprehensive understanding. c) Additional statistycal analisys through multivalued analysis and other techniques.

References

1. Geelan, J.: Twenty-one experts define cloud computing. Cloud Computing Journal,4, 1-5 (2009)
2. Fehling, C., Leymann, F., Retter, R., Schupeck, W., Arbitter, P.: Cloud Computing Patterns. Springer, Heidelberg (2014)
3. Gamez-Diaz, A., Fernandez, P., Ruiz-Cortes, A.: An Analysis of RESTful APIs Offerings in the Industry. In: Maximilien, M., Vallecillo, A., Wang, J., Oriol, M. (eds) Service-Oriented Computing. ICSOC 2017. Lecture Notes in Computer Science(), vol 10601. Springer, Cham (2017). doi:10.1007/978-3-319-69035-3 43
4. Fielding, R. T.: Architectural styles and the design of network-based software architectures. University of California, Irvine (2000)
5. Andreo, S., Bosch, J. (2019). API Management Challenges in Ecosystems. In: Hyrynsalmi, S., Suoranta, M., Nguyen-Duc, A., Tyrväinen, P., Abrahamsson, P. (eds) Software Business. ICSOB 2019. Lecture Notes in Business Information Processing, vol 370. Springer, Cham. doi:10.1007/978-3-030-33742-1 8
6. Brown, A., Fishenden, J., Thompson, M., Brown, A., Fishenden, J., Thompson, M.: API economy, ecosystems and engagement models. Digitizing Government: Understanding and Implementing New Digital Business Models, 225-236 (2014)
7. Bloch, J.: How to design a good API and why it matters. In: Companion to the21st ACM SIGPLAN Conference on Object-Oriented Programming Systems, Languages, and Applications - OOPSLA 06 (2006)
8. Brajesh, De: API management - an architect's guide to developing and managing APIs for your organization. Apress, Bangalore (2017)
9. Preibisch, S,: API Development. Apress (2018).

10. Gamez-Diaz, A., Fernandez, P., Ruiz-Cortes, A.: Governify for APIs: SLA-Driven ecosystem for API governance. In Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering, pp. 1120-1123 (2019)
11. Schultis, K.-B., Elsner, C., Lohmann, D.: Architecture challenges for internal software ecosystems: a large-scale industry case study. In: Proceedings of the 22Nd ACM SIGSOFT International Symposium on Foundations of Software Engineering, FSE 2014, pp. 542–552. ACM, New York (2014). doi:10.1145/2635868.2635876
12. Wilde, E., Amundsen, M.: The challenge of api management: API strategies for decentralized api landscapes. In Companion Proceedings of The 2019 World Wide Web Conference, pp. 1327-1328 (2019). doi:10.1145/3308560.3320089
13. Mathijssen, M., Overeem, M., Jansen, S.: Identification of practices and capabilities in API management: a systematic literature review. arXiv preprint arXiv:2006.10481 (2020). doi:10.48550/arXiv.2006.10481
14. Wohlin, C., Runeson, P., Ho¨st, M., Ohlsson, M. C., Regnell, B., Wesslén, A.: Experimentation in software engineering. Springer Science & Business Media (2012). doi:10.1007/978-3-642-29044-2
15. Fowler Jr, F. J.: Survey research methods. Sage publications (2013)
16. Kasunic, M.: Designing an effective survey. Software Engineering Institute, Pittsburgh (2005). doi:10.1184/R1/6573062.v1
17. Cochran, W. G.: Sampling Techniques (3rd ed.). John Wiley & Sons (1977)
18. Kish, L.: Survey Sampling. John Wiley & Sons (1965)
19. Cohen, J.: Statistical Power Analysis for the Behavioral Sciences (2nd ed.). Lawrence Erlbaum Associates (1988)
20. Rosenthal, R., Rosnow, R. L.: Essentials of Behavioral Research: Methods and Data Analysis (3rd ed.). McGraw-Hill (2008)
21. Ghazi, A. N., Petersen, K., Reddy, S. S. V. R., Nekkanti, H.: Survey Research in Software Engineering: Problems and Mitigation Strategies. In IEEE Access, vol. 7, pp. 24703-24718 (2019). doi:10.1109/ACCESS.2018.2881041
22. Espinha, T., Zaidman, A., Gross, H.: Web API growing pains: Stories from client developers and their code, 2014 Software Evolution Week - IEEE Conference on Software Maintenance, Reengineering, and Reverse Engineering (CSMR-WCRE), pp. 84-93 (2014). doi:10.1109/CSMR-WCRE.2014.6747228
23. RapidAPI: State of API Report 2022. RapidAPI (2023).
24. SmartBear: State of Software Quality - API: Latest Trends and Insights for 2023. Smartbear (2023).
25. Constanzo, M., Casas, S. I., Vidal, G. B., Cruz, D.: Usos y problemas de las APIsWeb en la República Argentina. Revista Tecnología y Ciencia, (44), 79–97 (2022). doi:10.33414/rtyc.44.79-97.2022

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Evaluación de requisitos de software utilizando GQM

Gladys Kaplan[1], Gerardo Riera[1], María Inés Bertín[1], Gabriel Blanco[1]

[1]Departamento de Ingeniería e Investigaciones Tecnológicas,
Universidad Nacional de La Matanza. San Justo, Buenos Aires
{gkaplan,griera,mbertin,gblanco}@unlam.edu.ar

Abstract. La Ingeniería de Requisitos ha puesto mucho énfasis en asegurar la calidad de los requisitos. A pesar de estos esfuerzos, estas descripciones siguen presentando serios problemas. La amplia utilización del lenguaje natural puede ser uno de los responsables. Se ha intentado mitigar estos problemas construyendo modelos previos, glosarios, etc., pero aún no se han obtenido los resultados esperados. En este artículo se presenta un método para evaluar las *características* de los requisitos y determinar su grado de comprensión. Para ello se utilizó un método cuantitativo Gold Question Metric que a través de preguntas permite evaluar internamente cada requisito. Para probar el método propuesto se utilizaron especificaciones tomadas directamente de la industria, las cuales eran consideradas correctas por sus autores. En todos los casos, muchos de estos requisitos presentaron serios problemas de descripción. Adicionalmente, la ejecución del método generó un conjunto de dudas que resultaron muy promisorias para corregir cada requisito.

Keywords: requisitos del software, características de los requisitos, verificación, Gold Question Metric (GQM).

1. Introducción

El objetivo de la Ingeniería de Software es construir software de la mayor calidad posible al menor costo [1]. Para ello es indispensable contar con una Especificación de Requisitos de Software (ERS) completa y correcta que permita asegurar el punto de partida de la construcción del software [2]. En 1981, Basili [3] encontró cerca de 88 errores en una ERS de 400 páginas para el proyecto "*A-7E Operational Flight Program*". Esta ERS había sido escrita por un grupo de expertos en especificación de requisitos. En el mismo camino, los estudios de Boehm [4][5] han demostrado que el impacto potencial de los requisitos mal formulados es sustancial. Boehm estudio como los requisitos, la especificación y los errores de diseño son los más numerosos en un sistema, con un promedio del 64 % en comparación con el 36 % de los errores de codificación. Para el presente artículo se analizó, entre otras, una ERS del proyecto "*Seguimiento de Alumnos"* de una Universidad, la que contó con 34 páginas donde se describieron 211 requisitos. En este caso también fueron escritos por expertos en requisitos. Se detectó que más de la mitad de los requisitos presentaban algún tipo de problema. Entre los problemas que pueden acarrear requisitos mal comprendido se encuentra satisfacer parcialmente las necesidades del cliente, multiplicar el costo de construcción, incumplir los tiempos estimados, etc. Se debe considerar que el 71.8% de las ERS existentes utilizan el Lenguaje Natural (LN) [6] para representar los requisitos, siendo esto una gran ventaja para mejorar la comunicación entre los stakeholders y los desarrolladores, pero una desventaja por las dificultades intrínsecas que tiene, como la

ambigüedad y la falta de precisión. Para mitigar estos problemas se han propuesto diferentes mecanismos como puede ser la utilización de glosarios [7]; verificaciones exhaustivas, por ejemplo, con Inspecciones [8]; la generación de modelos previos a la ERS (Casos de Uso, Escenarios) [9][10]; entre otros. Ben Achour et al. [11] determinó que la mitad de los Casos de Uso de un proyecto tenían errores basados en la falta de comprensión de la terminología utilizada. Por otro lado, también se presentan problemas cuando existen modelos previos como es el caso de los Escenarios [12][13]. Estos Escenarios tienen empotrados los requisitos los cuales se extraen de aquellos episodios donde el actor *sistema* tiene un rol. Al especificar los requisitos desde los Escenarios, estos aparecen en la ERS en el mismo orden en cual fueron extraídos y muchos de ellos suelen comprenderse sin dificultad mientras se conserven en la misma secuencia. Esto se debe a que en los requisitos anteriores existe información que lo completa. Pero cuando esos requisitos se separan, por ejemplo, al ordenarlos por algún ítem de *importancia,* pierden contexto y pueden llevar a interpretaciones erróneas. Si bien este problema puede ser subsanado retornando al modelo que le dio origen, requiere de un esfuerzo adicional con un alto riesgo de no hacerlo e interpretar equivocadamente el requisito. Por otro lado, quienes son responsables de las ERS suelen no ser conscientes que existen estos problemas y se valida con el cliente-usuario y se construye el sistema de software con requisitos improcedentes.

En el presente artículo se propone medir cada requisito en particular tomando en cuenta las *características* que lo definen. A tal efecto se utilizó la clasificación de Wiegers [14] que establece las siguientes características: necesario, correcto, conciso, completo, consistente, no ambiguo, verificable, trazable y modificable. Para determinar el grado de cumplimiento se definió una métrica cuantitativa para cada requisito en particular utilizando el Método Goal Question Metric (GQM) [3]. Se construyó un conjunto de preguntas que, además de medir, permite conocer con exactitud el problema de descripción existente, por lo que actúa como una suerte de verificación que permite la rápida corrección del requisito. Cabe destacar que este método es útil para trabajar con cada requisito en particular, pero para liberar la ERS se deben evaluar las características y atributos de la especificación, pero este punto no es tratado en el presente artículo.

El artículo está organizado de la siguiente manera: en la sección 2 se explica el modelo GQM; en la sección 3 se describen las características con las cuales se construye el modelo propuesto; en la sección 4 se define el modelo GQM para evaluar las características de cada requisito; en la sección 5 se analizan los resultados obtenidos y en la sección 6 las conclusiones y trabajos futuros.

2. Goal Question Metric (GQM)

El modelo GQM fue desarrollado por Basili [3] para medir diferentes metas de una organización y saber si las mismas fueron alcanzadas. Para ello, primero debe especificar las metas, rastrear esos objetivos a los datos y finalmente, proporcionar un marco para interpretar los datos en relación con el objetivo. El método se lo utiliza buscando una mejora evolutiva de la calidad. El resultado de la aplicación del enfoque GQM es la especificación de un sistema de medición dirigido a un conjunto particular de problemas y un conjunto de reglas para la interpretación de los datos de medición.

Para Basili una medición puede ser más satisfactoria si es diseñada teniendo en cuenta las metas, y las preguntas ayudan a medir si se está alcanzando en forma exitosa la meta definida. Se busca mejorar la calidad y confiabilidad reduciendo costos, riesgos y mejorando tiempos.

El modelo de medición GQM tiene tres niveles:

1) **Nivel conceptual (OBJETIVO/GOAL)**

Un objetivo se define para un objeto, por una variedad de razones, con respecto a varios modelos de calidad, desde varios puntos de vista, en relación con un ambiente particular.

Los objetos de medición son:

Productos: Artefactos, entregables y documentos que se producen durante el ciclo de vida del sistema; Por ejemplo, especificaciones, diseños, programas, conjuntos de pruebas.

Procesos: actividades relacionadas con el software normalmente asociadas con el tiempo; P.ej. especificar, diseñar, probar, entrevistar.

Recursos: Elementos utilizados por los procesos para producir sus salidas; P.ej. personal, hardware, software, espacio de oficina.

2) **Nivel operativo (PREGUNTA/QUESTIONS)**

Se utiliza un conjunto de preguntas para caracterizar la forma la evaluación/logro de una meta específica que se va a realizar en base a algún modelo caracterizador. Las preguntas tratan de caracterizar el objeto de medición (producto, proceso, recurso) con respecto a un problema de calidad y determinar su calidad desde el punto de vista seleccionado.

3) **Nivel cuantitativo (MÉTRICA/METRIC)**

Un conjunto de datos está asociado con cada pregunta para responderla de forma cuantitativa.

Los datos pueden ser:

Objetivo: Si dependen únicamente del objeto que se está midiendo y no en el punto de vista desde el que se toman; Por ejemplo, el número de versiones de un documento, horas del personal dedicadas a una tarea, tamaño de un programa.

Subjetivos: Si dependen tanto del objeto que se está midiendo como el punto de vista desde el que se toman; Por ejemplo, legibilidad de un texto, nivel de satisfacción del usuario.

3. Características de los requisitos del software

Para construir el método fue necesario comprender cuál era el problema real. Para ello se examinaron algunos requisitos al azar y simplemente se los leyó. Durante este proceso aparecieron muchas dudas que no permitían comprender en su totalidad el servicio descripto. Para ilustrar este procedimiento se describen algunos ejemplos:

Requisito #1: *"El sistema debe asignar un número identificatorio a cada carrera".* Observación: ¿Cómo se compone ese número idenficatorio?

Requisito #2: *"Se permitirá el registro de pedidos de compra con datos obligatorios incompletos, los cuales podrán completarse posteriormente modificando el pedido. Antes de poder aprobarse los datos del pedido deben estar completos".* Observación: ¿Cómo se puede aceptar datos obligatorios incompletos, si son obligatorios? ¿Para aprobar el pedido deben estar completos los obligatorios o todos los datos? Además de las dudas, se puede desagregar en varios requisitos.

Requisito #3: *"El sistema debe permitir imprimir etiquetas de manera unitaria".* Observación: ¿se debe imprimir etiquetas de un solo producto o imprimir una única etiqueta de un único producto?

Requisito #4: *"El sistema deberá generar reportes de manera semanal, mensual y anual, según sea la exigencia".* Observación: ¿Cómo se determina la exigencia?

Las dudas u observaciones que aparecieron en los requisitos analizados se clasificaron según las *características* de Wiegers [12] en conciso, necesario, verificable, correcto, completo, no ambiguo, trazable, consistente y modificable. Pero algunas observaciones no pudieron incorporarse a esta clasificación. Por lo tanto, se agregaron las siguientes tres características:

- Indivisible.
- Autónomo.
- Punto de vista.

Un requisito es indivisible cuando no se puede extraer más de una funcionalidad del mismo. Su granularidad es la menor. Un requisito es autónomo cuando se lo puede comprender por sí mismo, sin depender de otros requisitos. Esto no se relaciona con las dependencias entre requisitos sino con la comprensión del servicio que describe. Finalmente, el punto de vista es para homogeneizar las descripciones, de esta manera todos los requisitos tienen la misma perspectiva y esto facilita la lectura y por consiguiente, su comprensión. En general para una ERS la perspectiva es la del producto ("El sistema debe …"), para las Historias de Usuario es la del usuario, etc. En la Fig.1 se puede observar la clasificación completa utilizada para crear el modelo GQM.

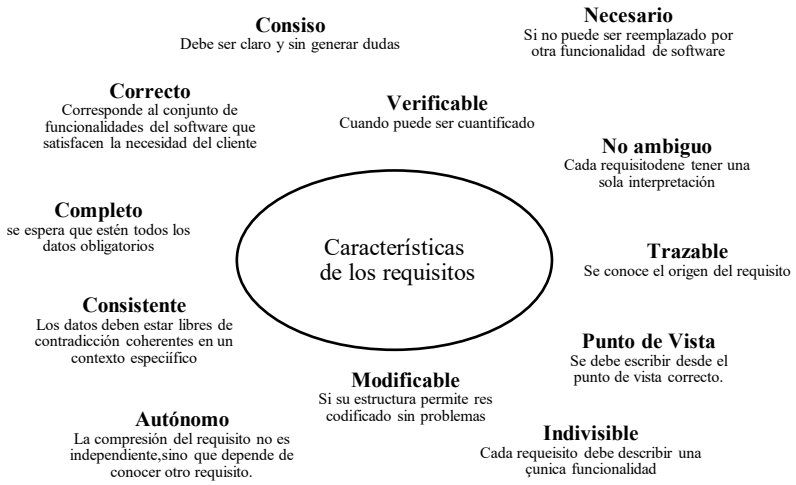


Fig. 1. Características de los requisitos

4. Modelo GQM para evaluar la calidad de los requisitos

Con todas las características de la sección anterior se construyó el modelo GQM definiendo los tres niveles. A continuación, se describe cada uno:

1) Nivel conceptual (OBJETIVO / GOAL)
   Objetivo: determinar la calidad de cada requisito de software.

2) Nivel operativo (PREGUNTA / QUESTIONS)
   Tipo de Respuesta: SI/NO

La Tabla 1 muestra las preguntas para cada característica y el peso asignado a cada una, que puede ser 0 o 1. En este caso solo se definió una pregunta por cuestiones de espacio, pero es posible profundizar en algunas características repartiendo el peso entre todas las preguntas formuladas.

Pregunta y **Peso** de cada característica:

Tabla 1. Nivel Operativo del GQM

Características	Preguntas	Peso en la fórmula
Correcto	**(P1)** ¿está dentro del alcance del Objetivo del Sistema?	RTA: NO=1
Conciso	**(P2)** ¿Es de fácil comprensión? (se puede comprender en una primera lectura)	RTA: NO=1
Necesario	**(P3)** ¿Puede ser reemplazado por otra funcionalidad?	RTA: SI=1
Verificable	**(P4)** ¿Están los parámetros/valores que permiten aplicar un método cuantificable?	RTA: NO=1
Completo	**(P5)** ¿Están los datos necesarios para implementar el requisito?	RTA: NO=1
Consistente	**(P6)** ¿Existe alguna contradicción en el propio requisito?	RTA: SI=1
Autónomo	**(P7)** ¿Alcanza con analizar este requisito solo? (sin conocer otros requisitos)	RTA: NO=1
Indivisible	**(P8)** ¿La realización del requisito corresponde a una única funcionalidad del software?	RTA: NO=1
Punto de Vista	**(P9)** ¿Está escrito desde la perspectiva correcta?	RTA: NO=1
No ambiguo	**(P10)** ¿Puede tener más de una interpretación?	RTA: SI=1
Trazable	**(P11)** ¿Se comprende el origen del requisito?	RTA: NO=1
Modificable	**(P12)** ¿Es posible de ser modificado forma fácil, completa y consistente?	RTA: NO=1

3) Nivel cuantitativo (MÉTRICA / METRIC)

Fórmula: 0 <= (∑ preguntas) / Cantidad de preguntas <= 1

La misma fórmula es utiliza para evaluar las características de un requisito y luego, se puede aplicar la misma fórmula para evaluar todo el requisito.

Evaluación:

La Tabla 2 muestra los rangos que permiten evaluar cada característica y luego, cada requisito. Puede observarse que estos valores pueden ajustarse rápidamente para contextos más críticos o particulares.

Tabla 2. Nivel Cuantitativo del GQM

Rangos x CARACTERÍSTICA		
Aceptado	A revisión	Rechazado
0	>=0.1 y <=0.99	=1

Rangos x REQUISITO		
Aceptado	A revisión	Rechazado
0	>=0.1 y <=0.39	>=0.4

5. Análisis Cuantitativo

Para realizar las pruebas del método se utilizaron cinco casos reales:

1) ERS#1 - Sistema de Organización Eventos
2) ERS#2 - Sistema de Gestión de Títulos
3) ERS#3 - Sistema de Rematriculación
4) ERS#4 - Sistema de Control de Calidad
5) ERS#5 – Gestión de Tiendas

En la ERS#1y ERS#5 los requisitos fueron descriptos directamente desde el contexto, mientras que la ERS#2, ERS#3 ERS#4 fueron obtenidas desde el modelo Escenarios, donde los requisitos están empotrados en los episodios. En estos casos, los requisitos extraídos tienen la característica de no presentar conflictos ya que se obtienen desde un modelo verificado y validado. Además, cuentan con un glosario, el cual se construye muy tempranamente en el proceso de requisitos utilizado.

ID_REQUISITO: #4
Descripción: El sistema debe proponer al alumno diferentes opciones de inscripción de acuerdo a si debe recursar materias o no. Si recursa, el sistema debe obligar a definir un turno para estas materias y luego completar las restantes. Si no recursa, el sistema debe permitir que elija el turno y las materias troncales, MOFE, MOGE y obligaciones académicas según corresponda a su carrera, ciclo y a la carga existente de la oferta académica.

Característica	Peso	Observaciones
Correcto	0	
Conciso	0	
Necesario	0	
Verificable	0	
Completo	1	No se entiende que se va a controlar en la prematriculación. Tampoco que es “periódico” (cuánto tiempo?)
Consistente	0	
Indivisible	1	Se habla de un proceso automático de pre-matriculación, y además de un envío de mail informando problema y posible solución. Al menos son 2 funcionalidades.
Autónomo	1	No se sabe cómo es el proceso de pre-matriculación automático.
Punto de Vista	0	
No ambiguo	0	
Trazable	1	
Modificable	1	
TOTAL x Requisito	**5**	

Fig. 2. Ejemplos del Nivel Operativo

Cabe aclarar que algunas características tienen una relación muy estrecha, por ejemplo cuando un requisito no está completo genera ambigüedad y a su vez hace dificultosa la trazabilidad. En estos casos se sugiere responder cada pregunta de manera independiente.

A modo de ejemplo, en la Fig.2 se muestra la evaluación de características realizada sobre el Requisito #4 de la ERS#3. Cabe destacar que, en todos los casos, los requisitos han sido tomados textualmente de la especificación correspondiente. La columna "Peso" contiene 0 o 1 de acuerdo a la respuesta de la pregunta de cada característica (ver Fig. 2). En la columna "Observaciones" se describen las dudas que aparecen durante el análisis. Esta columna puede convertirse en un mecanismo de corrección valioso, ya que puede ser entregado a los autores para que se enfoquen directamente en cómo mejorar cada descripción. Finalmente, se suma la columna "Peso" para obtener el "Total x Requisito", este valor indica la calidad del mismo. Este total se utiliza en la columna "Fórmula" de la Tabla 3.

Tabla 3. Nivel Cuantitativo para la ERS#3

Id. Requisito	Fórmula *Total x Requisito / Cantidad de requisitos*	Decisión por requisito *Aceptado = 0 A revisión >= 0.1 y <= 0.49 Rechazado >= 0.5*
#1	1 / 10 = 0,1	A revisión
#2	7 / 10 = 0,7	**RECHAZADO**
#3	2 / 10 = 0,2	A revisión
#4	5 / 10 = 0,5	**RECHAZADO**
#5	6 / 10 = 0,6	**RECHAZADO**
#6	5 / 10 = 0,5	**RECHAZADO**
#7	4 / 10 = 0,4	A revisión
#8	0 / 10 = 0	ACEPTADO
#9	2 / 10 = 0,2	A revisión
#10	2 / 10 = 0,2	A revisión

Como se mencionó al comienzo de esta sección, se evaluaron cinco casos obteniendo los siguientes resultados finales, tal como se muestra en la Tabla 4.

Tabla 4. Resumen de los requisitos

CASOS	Total de Requisitos	Cant. Aceptados	Cant. A Revisión	Cant. Rechazados
ERS#1	11	3	2	6
ERS#2	18	5	8	5
ERS#3	10	1	5	4
ERS#4	18	6	7	5
ERS#5	21	4	8	9

La suma de todos los casos analizados alcanza a 78 requisitos de los cuales solo 19 han sido aceptados sin ninguna observación.

6. Conclusiones y Trabajos Futuros

En este artículo se presenta un método para determinar cuantitativamente la comprensión de cada requisito. Para ello se utiliza el modelo GQM que permite determinar qué características son satisfechas y cuáles no. Como ya se mencionó, las cinco ERS utilizadas han sido tomadas de casos reales, demostrando cómo se está trabajando en la industria con respecto a la especificación de requisitos. En este mismo camino, es importante destacar que cuando se consultó a las empresas acerca de sus documentos de especificación la respuesta fue alarmante, ya que muchas de ellas no contaban con documentos respaldatorios. En algunos casos lo justificaron con que en la actualidad solo hacen mantenimiento del software, en otros casos trabajan con el boca a boca, etc. Cabe destacar tambien la importancia de realizar procesos que garanticen la construccion de los requisitos. Esto puede observarse en la Tabla 4 en los resultados finales, donde las ERS#1 y ERS#5 que fueron realizadas directamente con el usuario, generan más dudas que cuando se construyen desde modelos previos. Cabe destacar que en estos casos la experiencia del ingeniero de requisitos influye directamente en los resultados.

Tabla 5. Porcentaje de características incumplidas

Característica	Cantidad Requisitos con defectos	Porcentaje
Completo	24	31.90
No ambiguo	12	15.38
Autónomo	7	8.98
Indivisible	6	7.69
Consistente	5	6.41
Punto de vista	1	1.28
Conciso	1	1.28
Verificable	1	1.28
Modificable	1	1.28
Correcto	1	1.28
Trazable	0	0
Necesario	0	0
Total	**59**	**76.76**

Como resultado final se presentan los datos en la Tabla 5. Para construir esta Tabla se aplanaron todos los requisitos (78 requisitos) y se agruparon por característica. Se puede observar que el problema de la incompletitud se ha presentado en 24 requisitos, determinando que el 31.90% de los problemas radican en esta característica. Le sigue en importancia la ambiguedad, luego la autonomía, la consistencia y la indivisibilidad y finalmente, el punto de vista y lo conciso. La característica autónomo se da con mayor frecuencia cuando existen modelos previos a la ERS los cuales tienen los requisitos empotrados, ya que la secuencia en el cual aparecen en el modelo suele ser el mismo que en la ERS.

Estos resultados corroboran que siguen existiendo especificaciones con serios problemas de comprensión.

Como trabajo futuro se espera analizar más casos e incluir *Historias de Usuario* y *Casos de Uso*. Si bien las pruebas realizadas sobre ERS son iniciales, los resultados

obtenidos son muy promisorios. También se espera realizar mediciones en la industria y estudiar cómo responde como mecanismo de aprendizaje.

Referencias

1. Ian Sommerville, "Ingeniería de Software", 10th Edition, ISBN 10 9332582696, 2017
2. J. Dick, E. Hull, and K. Jackson (2017), Requirements Engineering". 4ta Edition, Springer, ISBN:9783319610733, 3319610732.
3. V.R. Basili (1992) "Software Modeling and Measurement: The Goal Question Metric Paradigm,", Computer Science Technical Report Series, CS-TR-2956 (UMIACS-TR-9296), University of Maryland, College Park, MD.
4. Boehm, B. (1984) "Model and metrics for software management and engineering", IEEE Computer Society Press, pp. 4-9
5. Birihanu, Ermiyas. (2018). "Analysis of Software Quality Using Software Metrics".
6. International Journal on Computational Science & Applications. 8. 11-20. 10.5121/ijcsa.2018.8502.
7. Berry D. (2004) "Requirements Ambiguity", en el libro "Perspectives on Software Requirements", Kluwer Academic Publishers, Ed. Leite, J.C.S.P., Doorn, J.H. EEUU, ISBN: 1-4020-7625-8, Capítulo 2.
8. Ryan Mike, Wheatcraft, Louis, Dick, Jeremy and Zinni, Rick. (2015). On the Definition of Terms in a Requirements Expression. INCOSE International Symposium. 25. 10.1002/j.2334-5837.2015.00055.x.
9. Kaplan Gladys (2022) "Proceso de Requisitos Validado Empíricamente". Tesis doctoral de la UNLP, Capitulo 13.
10. Del Águila Cano, I. M. (2022). "Fundamentos de Ingeniería de los Requisitos". Editorial Universidad de Almería. ISBN: 9788413511610, 8413511615, pp 68-77.
11. Jacobson, I., Christerson, M., Jonsson, P., Overgaard, G.,Reading, (1992) "Object-oriented
12. Software Engineering - A Use Case Driven Approach", MA: Addison Wesley, Nueva York: ACM Press.
13. Ben Achour, C., Rolland, C., Maiden, N.A.M., Souveyet, C. (1999) "Guiding Use Case Authoring: Results of an Empirical Study", International Symposium On Requirements Engineering (RE´99), Limerick, Irlanda, IEEE Computer Society Press, pp.36-43.
14. Hadad G., Doorn J. y Kaplan G. (2009) "Explicitar Requisitos del Software usando escenarios", WER 09, IX Workshop de Engenharia de Requisitos.
15. Leite, J.C.S.P., Hadad, G.D.S., Doorn, J.H., Kaplan, G.N. (2000), "Scenario Construction Process", Requirements Engineering Journal, Springer-Verlag London Ltd., Vol.5, Nº1, pp. 38-61.
16. K.E. Wiegers, (1999) "Writing Quality Requirements – Process Impact". Disponible en: http://www.processimpact.com/articles/qualreqs.pdf

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Estudio de Técnicas de Procesamiento de Lenguaje Natural en la Validación de Requerimientos

Sonia Santana[1] [0000-0003-2163-120X],Lucrecia Perero[1] [0000-0002-7422-9479], Noelia Rodriguez[1] [0009-0009-4981-6792]

Leandro Antonelli[2] [0000-0003-1388-0337],Alejandro Fernandez[2] [0000-0002-7968-6871]

[1] Facultad de Ciencias de la Administración - Universidad Nacional de Entre Ríos
{sonia.santana, ramona.perero}@uner.edu.ar, romi2022@gmail.com

[2] Laboratorio de Investigación y Formación en Informática Avanzada (LIFIA),
Facultad de Informática, Universidad Nacional de La Plata
{alejandro.fernandez, leandro.antonelli}@lifia.info.unlp.edu.ar

Abstract. La validación de los requerimientos es una tarea fundamental en cualquier proyecto de Ingeniería de Software. La calidad de los requerimientos tiene un impacto significativo sobre los artefactos durante el desarrollo del software, debido a que los requerimientos se expresan generalmente en lenguaje natural, lo cual puede ocasionar distintos tipos de defectos como la ambigüedad, integridad y atomicidad. Por otra parte, el principal objetivo de la validación de los requerimientos es confirmar que los requerimientos especificados sean representaciones de las necesidades y expectativas de los usuarios y deben ser completos, correctos y consistentes entre otras características, evaluar estos atributos de calidad puede resultar una tarea difícil. En este trabajo se muestra el relevamiento de las principales tendencias de las técnicas de procesamiento de lenguaje natural (PLN) en la validación de requerimientos publicados en los últimos 13 años. Se seleccionaron y analizaron a partir de un enfoque basado en un esquema de caracterización para conocer información asociadas a los mismos.

Keywords: Ingeniería de requerimientos, validación de requerimientos, lenguaje de procesamiento natural, técnicas de procesamiento de lenguaje natural

1. Introducción

El objetivo de la validación de requerimientos es certificar que los requerimientos documentados sean un reflejo del sistema del mundo real para el cual se va a desarrollar el software. Las especificaciones de requerimientos representan las expectativas, necesidades y deseos de los interesados del sistema propuesto.

En el marco de la Ingeniería de Requerimientos la validación de los requerimientos es una tarea fundamental en cualquier proyecto de Ingeniería de Software y debe ser un proceso continuo en el ciclo de vida del desarrollo del sistema. Los defectos de los requerimientos se encuentran entre las principales causas del fracaso de los proyectos [1][2]. Un motivo importante es que los requerimientos están generalmente escritos en lenguaje natural. Esta técnica se utiliza desde los primeros tiempos de la ingeniería de software y aún hoy es una de las más utilizadas [3], por lo tanto, los problemas del lenguaje natural son inherentes a los requerimientos [4].

La comprensión del lenguaje natural es el área principal en el campo del PLN [5]. Además, el PLN ocupa un lugar destacado en la Ingeniería de Requerimientos (IR),

particularmente cuando se deben diseñar sistemas [6]. En esa situación crítica, las barreras de comunicación e idioma entre los clientes y el equipo del proyecto causan una gran cantidad de problemas que reducen la calidad de la aplicación y el seguimiento de los requerimientos. [6].

Este trabajo se enfoca en las principales tendencias de las técnicas de PLN en la validación de requerimientos del software y presenta un análisis de enfoques para conocer información asociadas a los mismos.

El resto del artículo está organizado de la siguiente manera. La sección 2 describe una revisión de la literatura para identificar los distintos enfoques de las técnicas relacionadas con el PLN en la validación de requerimientos. La sección 3 describe un análisis que consiste en obtener información de los enfoques sobre las características asociadas a técnicas de PLN en la validación de requerimientos. Además, se presentan los resultados del análisis comparativo realizado a los enfoques, donde se identificaron las fortalezas y debilidades de las técnicas de PLN en la validación de requerimientos. La sección 4 discute los resultados obtenidos.

Finalmente, la sección 5 expresa las conclusiones y trabajos futuros.

2. Revisión bibliográfica

En esta sección se realizó un proceso de tres fases: búsqueda, selección y evaluación de trabajos para una revisión bibliográfica de enfoques relacionados con el PLN en la validación de requerimientos. La pregunta a responder para el proceso de revisión es ¿Cuál es la tendencia de las prácticas relacionadas de PLN en la validación de requerimientos? Para las búsquedas se desarrollaron frases, en inglés y español, con el fin de encontrar respuestas a la pregunta inicial, que son las siguientes:

- (“requirements”) y (“validation” o “validate” o “validity”) y (“technique”) y (“natural language” o “NLP” o “natural language processing”)
- ("requisitos" o “requerimientos”) y ("validación" o "validar" o "validez") y ("técnica") y (“lenguaje natural” o “PLN” o “procesamiento lenguaje natural”)

Las búsquedas se realizaron en artículos publicados en las editoriales de información IEEE, Springer y ACM Digital Library, publicados entre 2010 y 2023.

Los criterios de inclusión y exclusión son los siguientes: a) Los artículos seleccionados deben estar directamente relacionados con el tema de PLN en la validación de requerimientos en el área de IR. b) Los artículos no deben discutir la validación de requerimientos en la fase de prueba del software desarrollado o en la implementación en prueba con respecto a los requerimientos. c) Los artículos no deben utilizar términos como "Revisión sistemática de la literatura", "Revisión de la literatura”, “Análisis sistemático”.

Luego, se aplicaron los criterios de evaluación presentados en la Tabla 1. La lista de criterios de evaluación fue adaptada de [7] y permite seleccionar los artículos finales en base a cuatro criterios que se enumeran.

Tabla 1. Criterios de evaluación.

Sección	*Criterios*
Introducción	1. ¿La introducción proporciona una descripción general del aporte para PLN en la validación de requerimientos? 2. ¿Está claramente definido el propósito / objetivo de la investigación?
Metodología	3. ¿Está claramente definida la metodología de investigación? 4. ¿Se establece una sucesión clara de pruebas a partir de las observaciones a las conclusiones? 5. ¿Se indican claramente los procedimientos de análisis?
Resultados	6. ¿Se indica adecuadamente el caso y su contexto? 7. ¿El estudio de caso se basa en la teoría y está vinculado a la literatura? 8. ¿Están los hallazgos claramente establecidos? ¿Los resultados ayudan a resolver los problemas de validación de requerimientos?
Conclusión	9. ¿Contiene conclusiones claras, implicaciones para la práctica y la investigación futura? 10. ¿Las conclusiones, implicaciones para la práctica y futuras investigaciones son derivadas adecuadamente para su audiencia? 11. ¿Existen límites o restricciones impuestas a la afirmación de conclusión?

En la búsqueda inicial aplicando la estrategia de búsqueda se encontraron 6045 artículos, luego de eliminar los artículos duplicados y aplicar los criterios de inclusión y exclusión definidos se preseleccionaron 576 artículos. Después de leer los títulos, los resúmenes, palabras claves y las conclusiones se preseleccionaron 85 artículos relevantes. Por último, luego de aplicar los criterios de evaluación definidos en la Tabla 1 se preseleccionaron 32 documentos, de los cuales posteriormente se seleccionaron solo aquellos que aportan una técnica de PLN para el proceso de validación de requerimientos por su modo de pensamiento, modelamiento, conceptos y trabajo. El resultado se presenta en la Tabla 2.

Tabla 2. Artículos seleccionados de la revisión bibliográfica

Ref. Bibliog.	Ref. Estudio	Año	Nombre
[8]	T1	2011	Automatic Validation and Correction of Formalized, Textual Requirements
[9]	T2	2012	Using Norm Analysis Patterns for Automated Requirements Validation
[10]	T3	2018	Automated Validation of Requirement Reviews: A Machine Learning Approach
[11]	T4	2019	A Scalable Operational Framework for Requirements Validation Using Semantic and Functional Models

3. Análisis de técnicas

En esta sección se comparan las técnicas seleccionadas con un enfoque basado en un esquema de caracterización [12] para conocer sus características, necesidades de información y restricciones de las técnicas de PLN en la validación de requerimientos.

3.1 Definición del esquema de caracterización para analizar las técnicas.

Para la comparación de las técnicas de PLN en la validación de requerimientos se adaptó el esquema de categorización definido por Vegas [12]. El esquema presenta tres niveles: nivel operacional, nivel de uso y nivel histórico, cada uno contiene distintos atributos que definen los elementos de cada nivel. Fue desarrollado para recopilar información relevante sobre técnicas de prueba de software que contribuye a un repositorio donde almacena la descripción de cada técnica. En este contexto, el esquema fue adaptado al análisis propuesto, se eliminó el nivel histórico relacionado con el repositorio y se modificaron algunos atributos del nivel operacional y de uso.

Cuando el esquema es instanciado en una variedad de técnicas, proporciona a los desarrolladores suficiente información para poder seleccionar las técnicas más adecuadas en un proyecto determinado. Esto asegura que las decisiones que toman se basan en el conocimiento objetivo de las técnicas. La información sobre las técnicas se organiza en dos niveles, los mismos fueron adecuados para la comparación de técnicas para la validación de requerimientos:

Nivel operacional. Este nivel está relacionado con las condiciones óptimas de operatividad de la técnica, una vez determinadas las características dadas del entorno en el que se va a aplicar la técnica. Esto significa que puede o no ser adecuado aplicar una determinada técnica en función de los conocimientos, experiencia del personal y existencia de herramientas. Este nivel contiene cuatro elementos objetivo, técnica, personal y resultados que muestran la información disponible.

Nivel de uso. Este nivel especifica las experiencias de los sujetos en el uso de la técnica y contiene dos elementos: proyecto, que especifica información sobre proyectos en los que se utilizó la técnica y satisfacción, que especifica qué opinión merece la técnica entre las personas que la utilizaron.

La Tabla 3 muestra el contenido del esquema de categorización adaptado.

Tabla 3. Esquema de categorización.

Nivel	*Elemento*	*Atributo*	*Descripción*
Operativo	Objetivo	Función	Función de la técnica.
		Etapa	Etapa de desarrollo del software vinculada a la técnica.
		Elementos	Elementos del desarrollo del sistema vinculados
	Técnica	Comprensibilidad	Facilidad de la técnica.
		Entradas	Entradas requeridas de la técnica.
		Adecuación	Ajuste de la técnica en el caso de estudio.
		Identificación de datos	Esfuerzo para identificar los datos de la técnica.
		Dependencias	Relaciones de la técnica con otros recursos.
		Fuentes de información	Información sobre la técnica.
		Herramientas	Herramientas utilizadas por la técnica.

Nivel	*Elemento*	*Atributo*	*Descripción*
	Personal	Experiencias	Experiencia requerida para usar la técnica.
		Conocimientos	Conocimientos requeridos para utilizar la técnica
	Resultados	Entorno	Entorno proporcionado por el conjunto de casos de estudio de la técnica.
		Eficacia	Capacidad del conjunto de datos para detectar defectos con la técnica.
		Propiedades	Propiedades del modelo de requerimiento propuesta por la técnica.
Uso	Proyectos	Referencia	Proyectos donde se ha utilizado la técnica.
		Personal	Personal que ha utilizado la técnica.
		Lenguaje	Lenguaje utilizado por la técnica.
	Satisfacción	Opinión	Opinión general sobre la utilización de la técnica.
		Beneficios	Beneficios de usar la técnica.
		Problemas	Problemas con el uso de la técnica.

El análisis consiste en responder al esquema de categorización presentado en la Tabla 3. Las Tablas 4,5,6 y 7 presentan las respuestas para cada técnica.

Tabla 4. Análisis de la técnica T1

Nivel	*Elemento*	*Atributo*	*Descripción*
Operativo	Objetivo	Función	Busca patrones estructurales simples dentro del Gráfico de Sintaxis Abstracta (ASG) generado a partir de la especificación de requerimientos.
		Etapa	Etapa inicial del desarrollo del software.
		Elementos	Requerimientos formales de software.
	Técnica	Comprensibilidad	Medio.
		Entradas	Especificación de requerimientos
		Adecuación	Describe una jerarquía de funciones en el dominio automotriz mediante la búsqueda de patrones simples dentro del ASG generado a partir de la especificación de requerimientos.
		Identificación de datos	Medio.
		Dependencias	Se puede relacionar con modelos gráficos basados en el Modelo de Lenguaje Unificado (UML).
		Fuentes de información	https://ieeexplore.ieee.org/abstract/document/59544 53.
		Herramientas	Reclipse Tool Suite, detecta patrones de diseño estáticos y dinámicos utilizando ingeniería inversa con eclipse.

Nivel	*Elemento*	*Atributo*	*Descripción*
			https://www.researchgate.net/publication/22155383 3_Reverse_Engineering_with_the_Reclipse Tool Suite.
	Personal	Experiencias	Lenguaje Natural Controlado (CNL).
		Conocimientos	Técnica ASG.
	Resultados	Entorno	Industria Automotriz.
		Eficacia	Control de funciones de confort de vehículos. Se logró validar los requerimientos de las funcionalidades del bloqueo de puertas del sistema interconectando con las funcionalidades ya existentes.
		Propiedades	Inconsistencia, completitud y defectos en los requerimientos.
Uso	Proyectos	Referencia	Sistema eléctrico de vehículos.
		Personal	Ingeniero de Requerimientos.
		Lenguaje	Inglés.
	Satisfacción	Opinión	La técnica puede ser factible para otros dominios de aplicación.
		Beneficios	Verifica automáticamente si los requerimientos formalizados violan ciertas reglas, que consisten en restricciones lógicas y pautas para especificaciones de alta calidad. Además, ofrece operaciones para la corrección de requerimientos.
		Problemas	Es necesario tener un conocimiento previo de la técnica utilizada para poder adaptar los artefactos que contienen información específica del dominio.

Tabla 5. Análisis de la técnica T2

Nivel	*Elemento*	*Atributo*	*Descripción*
Operativo	Objetivo	Función	Aplica patrones de análisis semántico de normas para la validación de requerimientos.
		Etapa	Etapa inicial del desarrollo del software.
		Elementos	Requerimientos formales de software.
	Técnica	Comprensibilidad	Media
		Entradas	Especificación de requerimientos
		Adecuación	Propone generar las normas de una biblioteca para la validación de los requerimientos del sistema.
		Identificación de datos	Medio
		Dependencias	Casos de uso y modelado de procesos comerciales.
		Fuentes de información	https://ieeexplore.ieee.org/document/6359965

Nivel	*Elemento*	*Atributo*	*Descripción*
		Herramientas	CommonRules de IBM para el PLN. Analizador de Stanford para el análisis de oraciones y normas. https://nlp.stanford.edu/software/lex-parser.shtml Etiquetador de Stanford para el etiquetado de partes del discurso https://nlp.stanford.edu/software/tagger.shtm
	Personal	Experiencias	Lenguaje Natural Controlado (LNC).
		Conocimientos	Armado de patrones de normas comerciales.
	Resultados	Entorno	Educativo.
		Eficacia	No proporciona información.
		Propiedades	Defectos en los requerimientos.
Uso	Proyectos	Referencia	Sistema biblioteca.
		Personal	Analistas de requerimientos.
		Lenguaje	Inglés.
	Satisfacción	Opinión	La técnica se aplicó sobre un escenario de normas de biblioteca, lo que resultó relativamente simple, debido a que casi todos los actores estaban en conocimiento del dominio de la biblioteca.
		Beneficios	La técnica brinda una ventaja adicional sobre las revisiones e inspecciones periódicas, ya que genera inferencias que se pueden comparar con el resultado esperado de un escenario en el mundo real. Esto ayuda a solucionar cualquier discrepancia en los requerimientos en una etapa temprana en lugar de esperar a que la fase de prueba descubra los defectos, lo que reduce tanto el costo como el tiempo dedicado al desarrollo de software.
		Problemas	La técnica se vuelve confusa en oraciones complejas para identificar normas que no se expresan como patrones simples de "si-entonces".

Tabla 6. Análisis de la técnica T3

Nivel	*Elemento*	*Atributo*	*Descripción*
Operativo	Objetivo	Función	Aplica análisis semántico y minería de grafos para encontrar patrones comunes e identificar requerimientos propensos a errores.
		Etapa	En la etapa inicial del desarrollo del software.
		Elementos	Requerimientos formales de software.
	Técnica	Comprensibilidad	Media.
		Entradas	Especificación de requerimientos.

Nivel	*Elemento*	*Atributo*	*Descripción*
		Adecuación	**Menú interactivo de restaurante.** Valida el error del requerimiento verdadero como útil y el requerimiento falso positivo como no útil. **Sistema de control de estacionamiento**. Analiza las prioridades de los requerimientos y determinar el error del requerimiento verdadero y requerimientos falsos positivos luego del análisis semántico. **Sistema de organización de préstamos**. Clasifica los requerimientos para aislar los requerimientos erróneos de los requerimientos falsos positivos.
		Identificación de datos	Medio.
		Dependencias	No proporciona información.
		Fuentes de información	https://www.researchgate.net/publication/328313448
		Herramientas	Algoritmo LSA (Análisis Semántico Latente), determina matemáticamente temas dentro de un documento https://github.com/Ioana-P/pca_and_clustering_for_edu_purposes/blob/master/newsgroups_LSA.ipynb. Algoritmo LDA (Asignación de Dirichlet Latente), encuentra temas a los que pertenece el documento en función de las palabras que contiene. https://github.com/lda-project/lda
	Personal	Experiencias	Lenguaje Natural Controlado (LNC).
		Conocimientos	Análisis semántico y métricas de evaluación. Minería de grafos.
	Resultados	Entorno	Comercial.
		Eficacia	**Sistema interactivo de restaurante**. Los dos modelos de entrenamiento probados obtienen una tendencia similar a las métricas de evaluación, el primero con el 65% y el segundo con el 45%. **Sistema de control de estacionamiento.** Modelo de películas se desempeñó mejor que el modelo de revisiones. **Sistema de organización de préstamos**. Los clasificadores de conjuntos separaron las fallas de no fallas con una precisión muy alta, en algunos casos entre el 85% y el 89%.
		Propiedades	Defectos en los requerimientos.
Uso	Proyectos	Referencia	Sistema interactivo de restaurante. Sistema de control de estacionamiento. Sistema de organización de préstamos.
		Personal	Profesionales industriales. Ingeniero de requerimientos.
		Lenguaje	Inglés.
	Satisfacción	Opinión	No existe literatura reportada sobre el uso de análisis semántico y minería de grafos para mejorar la calidad de los requerimientos en lenguaje natural registrados en la especificación de requerimientos. El uso de técnicas LSA y LDA

Nivel	*Elemento*	*Atributo*	*Descripción*
			para escribir requerimientos de alta calidad es un enfoque novedoso.
		Beneficios	La solución de minería ayuda a identificar los requerimientos que están más conectados y pueden haber sido afectados por las correcciones durante el ciclo de inspección.
		Problemas	No proporciona información.

Tabla 7. Análisis de la técnica T4

Nivel	*Elemento*	*Atributo*	*Descripción*
Operativo	Objetivo	Función	Aplica modelos semánticos y encuentra similitudes o relaciones entre los requerimientos.
		Etapa	En cualquier etapa del desarrollo de software.
		Elementos	Requerimientos informales de software.
	Técnica	Comprensibilidad	Media.
		Entradas	Requerimientos informales de software y los conjuntos de datos de dominio relacionado.
		Adecuación	La técnica está en los primeros pasos del desarrollo se aplicó en un sistema de gestión de cursos donde los profesores pueden crear contenido en línea para los estudiantes y administrar cursos en línea.
		Identificación de datos	Medio.
		Dependencias	Metodología Scrum.
		Fuentes de información	https://dl.acm.org/doi/10.1145/3305160.3305166
		Herramientas	Algoritmos Naive Bayes y Support Vector Machine (SVM) e Integration Definition for Function Modeling (IDEF0)
	Personal	Experiencias	Lenguaje Natural Controlado (LNC).
		Conocimientos	Modelos semánticos.
	Resultados	Entorno	Educativo.
		Eficacia	La técnica ofrece una lista de probabilidades de más de 300 entradas por cada palabra en el vocabulario de requerimiento.
		Propiedades	Inconsistencia en los requerimientos.
Uso	Proyectos	Referencia	Sistema de gestión de cursos.
		Personal	Experto en ciencia de datos, ingenieros de software senior, clientes y gerente de proyecto de software senior.
		Lenguaje	Inglés.
	Satisfacción	Opinión	La técnica encuentra defectos en cualquier fase del desarrollo de software basado en los requerimientos textuales, los conjuntos de datos de dominio y los modelos semánticos.

Nivel	*Elemento*	*Atributo*	*Descripción*
		Beneficios	La técnica es independiente del dominio del software y del idioma de los requerimientos textuales.
		Problemas	La restricción de la técnica es el conjunto limitado de datos disponibles, por ejemplo, leyes, reglamento, estándares de la organización entre otros.

4. Resultados

Del análisis de los resultados obtenidos de los enfoques indicados en la Tabla 2 y los aspectos evaluados en los enfoques en las Tablas 4, 5, 6, y 7 se extraen diferentes características que son calificadas como puntos de valoración para los enfoques seleccionados. En la Tabla 8 se enumeran las características identificadas, en la cual se indica con el vocablo SI que el enfoque contribuye a la característica, mientras que el vocablo NO indica que el enfoque no contribuye a la característica.

Tabla 8. Comparación de los enfoques de validación de requerimientos.

Técnica / *Característica*	T1	T2	T3	T4
Requerimientos formales	SI	SI	SI	NO
Requerimientos informales	NO	NO	NO	SI
Especificación de requerimientos	SI	SI	SI	NO
Correctitud	SI	SI	SI	NO
Consistencia	SI	NO	NO	SI
Completitud	SI	NO	NO	NO
Análisis sintáctico	SI	NO	NO	NO
Análisis semántico	NO	SI	SI	SI
Algoritmos de aprendizaje automático	SI	SI	SI	SI
Uso de patrones	SI	SI	SI	NO
Ciclo de vida del software	NO	NO	NO	SI
Dominio específico	SI	SI	SI	NO
Uso de métricas	NO	NO	SI	NO
Experiencia en LNC	SI	SI	SI	SI
Participación cliente	NO	NO	NO	SI
Idioma específico	SI	SI	SI	NO

Del análisis comparativo realizado entre los enfoques seleccionados se extraen los siguientes resultados:

1. Las técnicas T1, T2, T3 validan requerimientos formales de software, mientras que T4 valida requerimientos informales de software. Las entradas necesarias para aplicar las técnicas T1, T2 y T3 es la especificación de requerimientos, mientras

que la técnica T4 se aplica en los requerimientos informales de software y los conjuntos de datos de dominio relacionado.

2. La técnica T1 aplica las propiedades de consistencia, completitud y correctitud en los requerimientos, la técnica T2 y T3 aplica la propiedad de correctitud en los requerimientos y finalmente la técnica T4 aplica la propiedad de consistencia en los requerimientos.
3. Las técnicas T2, T3 y T4 utilizan el análisis semántico, mientras que T1 utiliza el análisis sintáctico en la validación de requerimientos.
4. Las técnicas T1, T2, T3 y T4 emplean algoritmos de aprendizaje automático como herramientas para la validación de requerimientos y requieren personal con experiencia en Lenguaje Natural Controlado.
5. Las técnicas T1, T2, T3 validan las propiedades de los requerimientos utilizando patrones, mientras que T4 valida las propiedades de los requerimientos mediante la relación de categorías semánticas.
6. Las técnicas T1, T2 y T3 se utilizan en la etapa inicial del desarrollo del software, mientras que la técnica T4 se puede utilizar en cualquier etapa del desarrollo del software.
7. Las técnicas T1, T2 y T3 se desarrollaron para un dominio especifico, mientras que T4 es una técnica independiente del dominio del software.
8. La técnica T3 emplea métricas para evaluar los resultados de la validación de los requerimientos.
9. Las técnicas T1, T2, T3 y T4 requieren personal con experiencia en LNC para su implementación.
10. La técnica T4 es la única que incorpora al cliente en la validación de requerimientos.
11. Las técnicas T1, T2 y T3 se implementan en un idioma especifico de requerimientos, mientras que T4 es independiente del idioma.

5. Conclusiones y Trabajos Futuros

Este trabajo se ha focalizado en un proceso fundamental de la Ingeniería de Requerimientos: la Validación. Se han preseleccionado 32 trabajos para posteriormente centralizarse en 4 de ellos. Para responder a las preguntas de evaluación de las cuatro técnicas de PLN seleccionadas se identificaron las contribuciones de los trabajos en el proceso de Validación de Requerimientos.

Las técnicas T2, T3 y T4 se centran en el análisis semántico, es decir, en el significado contextual de las palabras en la validación de requerimientos, mientras que la técnica T1 en el análisis sintáctico, es decir, analiza la disposición de las palabras en una oración para que tengan sentido gramatical en la validación de requerimientos.

Adicionalmente, las técnicas T1, T2 y T3 validan la propiedad de correctitud de los requerimientos utilizando patrones en la especificación de requerimientos y se aplican en la etapa inicial del desarrollo del software. La técnica T4 valida la propiedad de consistencia de los requerimientos mediante la relación de categorías semánticas en todo el ciclo de vida del desarrollo del software, esto se debe a que la técnica es

independiente del dominio del software y del idioma de los requerimientos textuales. Además, hay que considerar que la técnica T4 está en los primeros pasos del desarrollo.

Las técnicas proporcionan diferencias en el enfoque de dominio en la validación de requerimientos con diversos grados de éxito. Este éxito depende de la naturaleza de la organización en sí y del conocimiento profundo del negocio para adaptar la técnica a las necesidades del negocio y del usuario.

Se evidencia el escaso uso de indicadores de rendimiento para el control y seguimiento de los defectos en los requerimientos, así como también objetivos a ser evaluados por dichos indicadores.

Todas las técnicas requieren personal con experiencia en LNC debido a que aplican algoritmos de aprendizaje automático en la validación de requerimientos, es decir, el personal debe tener conocimiento sobre lenguaje natural y la aplicación de restricciones de vocabulario, gramática y/o semántica.

Para fortalecer el proceso de validación de requerimientos, como trabajo futuro se propone crear un proceso de validación a partir de la técnica de PLN seleccionada, donde se definen las actividades del proceso y se describen objetivos, tareas, roles, productos de entradas y salidas para cada actividad utilizando un entorno colaborativo, que permitiría a las diferentes partes interesadas en un proyecto validar requerimientos. De esta manera, se reduciría el tiempo y esfuerzo necesarios para validar los requerimientos y se aseguraría que todos los stakeholders estén de acuerdo en cuanto a las entidades relevantes y requerimientos del proyecto.

Referencias

1. R. Chillarege, W. L. Kao, R. G. Condit: Defect Type and its Impact on the Growth Curve, in the 13th IEEE International Conference on Software Engineering, Austin, Texas, May 13-17, (1991).
2. Reyes Juárez Ramírez: Towards improving user interfaces: a proposal for integrating functionality and usability since early phases, IEEE, Indonesia, (2011).
3. Nikora Alen P.: Experiments in Automated Identification of Ambiguous Natural-Language Requirements, (2011).
4. D. Berry, E. Kamsties: The sintactically dangerous all and plural in specification, IEEE Software, pp. 55-57, (2005).
5. xAli Memon Kamran, Xiaoling Xia: Deciphering and Analyzing Software Requirements employing the techniques of Natural Language Processing, 4th International Conference on Mathematics and Artificial Intelligence (ICMAI 2019), Association for Computing Machinery, New York, NY, USA, 153–156, (2019).
6. S. Peshterliev, J. Kearney, A. Jagannatha, I. Kiss, S. Matsoukas: Active Learning for New Domains in Natural Language Understanding, (2018).
7. Nor Aiza Moketar, Massila Kamalrudin, Mohd Yusof Mokhtar, Sidek Safiah: A review on requirements validation for software development, Journal of Theoretical and Applied Information Technology, (2018).
8. J. Holtmann, J. Meyer, M. von Detten: Automatic Validation and Correction of Formalized, Textual Requirements, IEEE Fourth International Conference on Software Testing, Verification and Validation Workshops, Berlin, Germany, 2011, pp. 486-495, (2011).
9. R. Sharma, K. K. Biswas: Using norm analysis patterns for automated requirements validation, Second IEEE International Workshop on Requirements Patterns (RePa), Chicago, IL, USA, pp. 2328, (2012).

10. M. Singh: Automated Validation of Requirement Reviews: A Machine Learning Approach, IEEE 26th International Requirements Engineering Conference (RE), Banff, AB, Canada, pp. 460-465, (2018).
11. Issa Atoum: A Scalable Operational Framework for Requirements Validation Using Semantic and Functional Models, in Proceedings of the 2nd International Conference on Software Engineering and Information Management (ICSIM '19), Association for Computing Machinery, New York, NY, USA, 1–6, (2019).
12. S. Vegas, V. Basili: A characterisation schema for software testing techniques, Empirical Softw. Engg., vol. 10, no. 4, pp. 437–466, (2005).

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Calidad 4.0 en Ingeniería de Software: propuesta de un marco de trabajo

Kristian Petkoff Bankoff[0000-0001-6707-9268], Rocío Muñoz[0000-0001-6521-0568], Ariel Pasini[0000-0002-4752-7112], and Patricia Pesado[0000-0003-0000-3482]

Instituto de Investigación en Informática LIDI (III-LIDI)
Facultad de Informática, Universidad Nacional de La Plata, Argentina
{kpb,rmunoz,apasini,ppesado}@lidi.info.unlp.edu.ar

Resumen La Ingeniería de Software es una disciplina en constante evolución que se encuentra presente en todos los aspectos de la vida cotidiana. La calidad del software es un aspecto abordado desde hace tiempo, surgida de los mismos principios de la calidad en la industria manufacturera pero adaptada a las características propias del desarrollo de productos no tangibles. La industria ha evolucionado hacia una revolución digital que implica el uso de tecnologías emergentes como Internet de las cosas (IoT), inteligencia artificial y aprendizaje automático. La Calidad 4.0 adopta los principios de la Industria 4.0 y los aplica a la gestión de la calidad, con un enfoque basado en el análisis de grandes volúmenes de datos, el monitoreo en tiempo real y un mayor enfoque en el cliente mediante el uso de herramientas de comunicación y colaboración. Se propone un marco de trabajo que integra diferentes procesos, herramientas y técnicas de la Ingeniería de Software con el objetivo de habilitar la implementación de principios de la Calidad 4.0 en el desarrollo de software, tomando a DevOps como habilitador al ser un enfoque que promueve la automatización y la integración de herramientas.

Keywords: Quality 4.0, Software Engineering, DevOps

1. Introducción

El concepto de Calidad 4.0 se refiere a la integración de tecnologías digitales en los sectores industriales y de fabricación para potenciar la toma de decisiones basada en datos, el monitoreo en tiempo real y la integración de tecnologías como Internet de las cosas (IoT), inteligencia artificial y análisis de grandes volúmenes de datos[1].

El desarrollo de software es una parte crítica de la digitalización de la industria que promueve este enfoque [2] ya que es transversal a prácticamente todos los demás principios y características del mismo. Por otra parte, la gestión de la calidad en el proceso de desarrollo de software y en la del propio producto de software es de suma relevancia, lo que se demuestra en la diversidad de estándares disponibles y su aplicación en la industria [3].

En este artículo se propone un marco de trabajo que integra diferentes procesos, herramientas y técnicas de la Ingeniería de Software con el objetivo de habilitar la implementación de principios de la Calidad 4.0 en el desarrollo de software, utilizando como habilitador el concepto de DevOps [4].

El artículo se encuentra organizado de la siguiente manera: en la sección 2 se presentan los conceptos principales de Calidad 4.0, del proceso de Ingeniería de

Software, metodologías ágiles y la cultura DevOps; en la sección 3 se presentan las características que requiere la implementación del framework propuesto; en la sección 4 se describen las herramientas seleccionadas para cubrir los requerimientos propuestos y la forma en que se integraron; a continuación, en la sección 5, se resumen los resultados obtenidos, las conclusiones y los trabajos futuros.

2. Calidad 4.0 e Industria 4.0

El concepto de Calidad 4.0 se basa en la cuarta revolución industrial, también conocida como Industria 4.0, que se refiere a la integración de tecnologías digitales en los sectores industriales y de fabricación.

La Industria 4.0 aprovecha la automatización, los dispositivos IoT, la computación en la nube y los sistemas ciberfísicos para crear fábricas y procesos inteligentes y conectados. La Calidad 4.0 adopta los principios de la Industria 4.0 y los aplica a la gestión de la calidad, aprovechando las tecnologías emergentes, el análisis de datos y la transformación digital para impulsar la mejora continua y centrarse en el cliente en varias industrias, adaptable a la de servicios y a la ingeniería de software.

En esencia, la Calidad 4.0 enfatiza la toma de decisiones basada en datos, el monitoreo en tiempo real y la integración de tecnologías como IoT, inteligencia artificial y análisis de big data [5].

Al centrarse en las necesidades del cliente, involucrar a las partes interesadas en el proceso de desarrollo y adaptarse continuamente a los requisitos cambiantes, la Calidad 4.0 tiene como objetivo mejorar la calidad de los productos y servicios, optimizar los procesos e impulsar la innovación en la era digital.

2.1 Principios de la Calidad 4.0

Existen diversas propuestas para dividir la Calidad 4.0 en características, principios o ejes. Debido a que el concepto se apoya en la Industria 4.0 y en los procesos de producción de bienes tangibles, se hace a continuación una selección de los principios mejor alineados con la industria del software.

- Toma de decisiones basada en datos: aprovecha el análisis de datos, big data e inteligencia artificial para informar la toma de decisiones. En la industria del software, esto se traduce en el uso de monitoreo en tiempo real, modelos de predicción de errores y datos de comentarios de los clientes para guiar los procesos de desarrollo y optimizar la calidad del software.
- Mejora Continua: fomenta una cultura de mejora continua. Este principio se alinea con la metodología de desarrollo de software ágil, enfatizando el desarrollo iterativo, las retrospectivas y el aprendizaje de los comentarios de los clientes para mejorar los productos de software continuamente.
- Centrado en el cliente: comprender las necesidades y preferencias de los clientes es crucial para los equipos de ingeniería de software. Al involucrar a los clientes en todo el proceso de desarrollo y brindarles experiencias personalizadas, los productos de software pueden satisfacer mejor las expectativas de los usuarios.

- Integración de tecnologías emergentes: para la industria del software, abarcar tecnologías emergentes significa adoptar prácticas DevOps, integración continua e implementación para mejorar la colaboración y agilizar la entrega de software.

2.2 Relación entre el software y Calidad 4.0

El software sirve como un habilitador crítico para la implementación de Calidad 4.0 en todas las industrias. Dado que la Calidad 4.0 enfatiza la toma de decisiones basada en datos, el monitoreo en tiempo real y la integración de tecnologías emergentes, el software proporciona la infraestructura necesaria para recopilar, procesar y analizar grandes cantidades de datos. A la vez, permite la conectividad y la interoperabilidad de dispositivos y sistemas en el panorama de IoT, lo que permite el intercambio de datos entre los ámbitos físico y digital. El aprovechamiento de la inteligencia artificial y del aprendizaje automático para identificar patrones, predecir defectos y optimizar procesos en tiempo real aportan mejoras sustanciales a los procesos.

La flexibilidad y adaptabilidad del software son esenciales para implementar prácticas de mejora continua, desarrollo iterativo y enfoques centrados en el cliente. Uno de los principios sobre los que se basa la Calidad 4.0 es el desarrollo de aplicaciones de software ad-hoc para satisfacer las necesidades propias de la organización de manera específica y personalizada. A su vez, los principios y conceptos de la Calidad 4.0 tienen implicaciones significativas para la industria del software. Al aprovechar los datos y el análisis, los equipos de ingeniería de software pueden tomar decisiones informadas, identificar tempranamente defectos y mejorar la calidad general del software.

Por otra parte, la Calidad 4.0 se alinea con las metodologías ágiles, lo que promueve el desarrollo iterativo, la retroalimentación continua del cliente y la colaboración; esto permite que los equipos entreguen productos centrados en el cliente que satisfagan las necesidades cambiantes. Al adoptar enfoques centrados en el cliente e involucrar a las partes interesadas en todo el proceso de desarrollo, la ingeniería de software responde mejor a las necesidades y preferencias de los usuarios, lo que da como resultado mejores productos de software.

La Calidad 4.0 impulsa el desarrollo de software hacia prácticas basadas en datos, fomentando el uso de monitoreo en tiempo real, análisis predictivo y modelos de predicción de errores. Esas prácticas facilitan la supervisión continua de los sistemas de software y fomentan una cultura de mejora y aprendizaje continuos.

Los modelos de desarrollo de software ágiles proponen un proceso liviano que tiene como objetivo reducir los costos, potenciar la comunicación y la coordinación y permitir un enfoque adaptable y flexible ante los cambios en los requerimientos [6]. Estas características están directamente alineadas con principios fundamentales de la Industria 4.0 y sirve de base para la Calidad 4.0. Es por esto que a la hora de proponer un marco de trabajo para implementar el concepto de Calidad 4.0 a la Ingeniería de Software resulta conveniente seleccionar modelos de proceso mejor adaptados desde su concepción. Dos de los modelos ágiles más utilizados son Scrum y XP [7]

La intersección entre la Ingeniería de Software y la Calidad 4.0 puede ser llevada adelante a través de DevOps, que es un conjunto de prácticas, filosofías culturales y herramientas de colaboración que tienen como objetivo mejorar y optimizar el ciclo de

vida del desarrollo de software al fomentar la colaboración entre los equipos de desarrollo de software (Dev) y operaciones de TI (Ops). El objetivo de DevOps es automatizar y optimizar el proceso de creación, prueba, implementación y mantenimiento de aplicaciones de software, lo que da como resultado ciclos de desarrollo más rápidos, calidad mejorada y mayor eficiencia. Para esto define un proceso de integración continua (CI) donde los desarrolladores combinan con frecuencia sus cambios de código en un repositorio compartido. Este enfoque ayuda a identificar los problemas de integración de forma temprana, evitando la acumulación de defectos y reduciendo el riesgo de que los errores lleguen a producción. Al detectar problemas en una etapa temprana, se mejora la calidad general del software. Las pruebas automatizadas, incluidas las pruebas unitarias, las de integración y las de uso de extremo a extremo, se ejecutan automáticamente con cada confirmación de código, asegurando que los cambios no introduzcan regresiones o defectos, llevando así a un software más confiable y de mayor calidad.

El proceso de entrega continua (CD) implica que el software se lance a entornos de producción o ensayo de forma rápida y frecuente. La automatización de los procesos de implementación garantiza la coherencia y reduce las posibilidades de errores humanos, además de posibilitar una mejor experiencia del usuario con respecto a las actualizaciones del software. Por otra parte, el uso de la infraestructura como código (IaC), donde el aprovisionamiento y la configuración de la infraestructura se automatizan y versionan junto con el código fuente de la aplicación, reduce la desviación de la configuración y garantiza que el software se ejecute de manera consistente en diferentes entornos, permitiendo reorientar esfuerzos en otras áreas al mantener entornos de ejecución estables.

La naturaleza de integrar las operaciones de TI con todas las partes interesadas de los procesos de desarrollo requiere de la implementación de prácticas de comunicación mejorada y de responsabilidades compartidas. DevOps fomenta un ciclo de retroalimentación rápido entre desarrolladores, probadores y otras partes interesadas a partir de la automatización de la línea de trabajo y de la disponibilidad de datos. Esta retroalimentación constante permite una rápida identificación y resolución de problemas de calidad, lo que lleva a menores costos por errores y a potenciar la mejora del proceso de desarrollo.

Otro principio clave de DevOps es el monitoreo y análisis en tiempo real, que brinda información sobre el rendimiento del software, los patrones de uso y los posibles problemas, permitiendo a los equipos abordar de manera proactiva las inquietudes relacionadas con la calidad.

Las características descritas anteriormente permiten demostrar que DevOps adopta una cultura de mejora continua, en la que los equipos evalúan y optimizan periódicamente sus procesos. Este enfoque iterativo para el desarrollo y las operaciones contribuye a las mejoras continuas en la calidad del software. La integración continua, la entrega continua, las pruebas automatizadas y la configuración de entornos de ejecución como código permiten controlar y mejorar atributos de la calidad del software además del proceso en sí mismo.

3. Propuesta de framework para aplicar Calidad 4.0 en la Ingeniería de Software

La Calidad 4.0 propone un enfoque en el que la utilización de características de la Industria 4.0 se integran en todos los procesos de la organización. Cuando se analiza la aplicación de la Calidad 4.0 en el contexto de procesos de desarrollo de software se debe tener en cuenta que no todos los principios propios de la manufactura de bienes pueden ser trasladados de forma directa a esta disciplina, ya que la Ingeniería de Software no tiene como objetivo producir bienes tangibles, no persigue la producción en serie y tiene un ciclo de vida más allá de la entrega al cliente.

La propuesta de este trabajo se limita a los principios que mejor se adaptan a la industria del software, tomando un subconjunto relevante de los principios aplicables por cuestiones de extensión de la implementación.

- Trazabilidad: es una característica común a todas las características y funcionalidades del marco de trabajo que las actividades sean trazables y que el estado de los artefactos sea transparente y visible para todas las partes interesadas. La propuesta involucra el uso de contratos inteligentes para registrar los requerimientos del software y posteriormente toda acción que se provoque sobre los mismos, desde su planificación hasta la entrega, pasando por las diferentes etapas del desarrollo y prueba. El uso de blockchain y contratos inteligentes se propone como habilitador de la Calidad 4.0 por formar parte de los principios de la Industria 4.0.
- Automatización: cada una de las herramientas y procedimientos que se seleccionen debe permitir la integración automática por medio de API de manera de posibilitar la ejecución autónoma de los procesos alcanzados por el framework y, en especial, la producción de datos de trazabilidad que sean registrados por medio de contratos inteligentes. Esta característica está alineada con la cultura DevOps que se propone como habilitadora de la Calidad 4.0.
- Aseguramiento de la calidad: el abordaje debe realizarse integrando herramientas que permitan la automatización del aseguramiento de la calidad, obteniendo métricas del producto y del proceso por su propia ejecución y no por evaluaciones o inspecciones manuales. Las métricas y los indicadores clave de proceso (KPI) deben poder ser definidas por la organización de manera de que sean específicos para el proyecto y opcionalmente pueden ser adaptadas para satisfacer estándares de calidad como por ejemplo la familia de normas ISO 25000. Es deseable que los parámetros de las herramientas de análisis de calidad del producto sean fácilmente ajustables para permitir su adaptación a los cambios en el contexto.
- Gestión del ciclo de vida de los requerimientos: las metodologías ágiles promueven la flexibilidad, adaptabilidad y el uso de procesos más livianos; en este contexto, se debe permitir registrar los requerimientos de una forma ágil pero a la vez coherente y trazable. El ciclo de vida de los requerimientos quedará acotado a la creación, el debate técnico y funcional, la implementación, finalización, prueba y aceptación de cada requerimiento individual.
- Gestión de la configuración del software: esta característica consiste en controlar los cambios en el software a lo largo de su desarrollo, basado en el versionado del código, la definición de procedimientos para la integración de los cambios, el

control de entrega del software y el mantenimiento de la consistencia [8].

- Integración y adaptación de servicios y visualización de la traza: la integración de diferentes servicios y herramientas para propiciar un registro centralizado y trazable de las actividades del proyecto implican un esfuerzo adicional para sincronizar las diferentes etapas y artefactos y además mantener la coherencia entre los diferentes modelos de datos que utiliza cada aplicación del ecosistema [9]. La información de la traza se debe mostrar de una manera intuitiva al usuario, ya que es primordial involucrar al cliente de extremo a extremo en el proceso. Se definen dos niveles de visualización de trazabilidad: un nivel de planificación, orientado al cliente, y un nivel técnico, con mayor nivel de detalle, de manera que el usuario puede elegir el modo de visualización según sus necesidades.

4. Implementación

El framework propuesto fue evaluado en el contexto de un proyecto llevado adelante por un grupo de alumnos de una asignatura del tercer año de la Facultad de Informática de la Universidad Nacional de La Plata.

El proyecto consistió en el desarrollo de un sistema de gestión de turnos para un centro de vacunación. Los requerimientos del sistema fueron especificados utilizando historias de usuario y la implementación fue llevada adelante en un plazo de dos meses, divididos en tres sprints, basado en el modelo Scrum. La pila de producto y la gestión de los sprints fue realizada con la herramienta Pivotal Tracker. El código fuente del sistema fue alojado en un repositorio de GitLab y la integración continua fue realizada con GitLab CI, mientras que la herramienta de análisis de calidad del código utilizada fue SonarQube.

Para la integración del ecosistema de herramientas se desarrolló un agente ad-hoc que se encarga de la sincronización de los datos entre cada una estas, ya sea por medio de polling o por medio de webhooks según fuera necesario según las características de cada servicio. La Figura 1 muestra el diagrama de actividades alcanzadas por el framework propuesto.

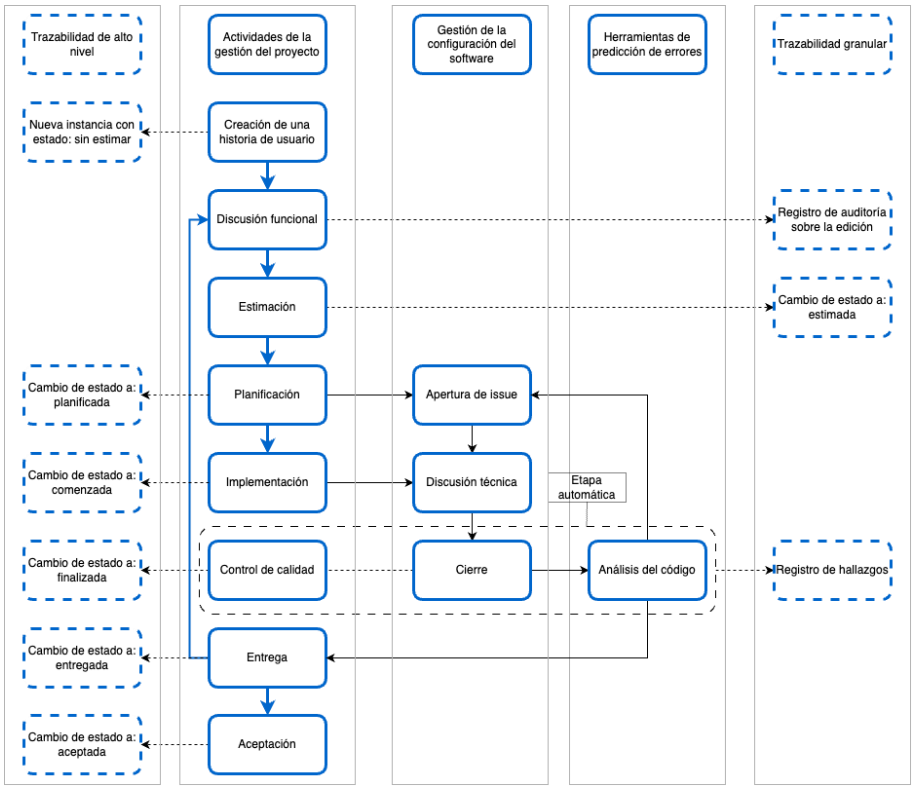


Figura1. Diagrama de actividades alcanzadas por el framework propuesto

Las columnas representan a cada uno de los siguientes componentes:

- Trazabilidad de alto nivel: registros de actividad que son de mayor atención para el cliente y para el seguimiento en general del proceso de desarrollo. Se incluyen registros producidos por la aplicación Pivotal Tracker, como la creación de la historia de usuario y los cambios de estado. Esta característica es implementada con contratos inteligentes que registran las transacciones en una red Hyperledger Fabric y una aplicación web de visualización de la traza desarrollada específicamente para este propósito y ajustada al modelo de datos utilizado.
- Actividades de gestión del proyecto: corresponde a la creación de la pila de producto mediante la especificación con historias de usuario y discusión a través de comentarios, la planificación de la pila del sprint y el seguimiento del estado de cada requerimiento funcional hasta su entrega en la herramienta Pivotal Tracker, enmarcada en el modelo Scrum.
- Gestión de la configuración del software: en esta implementación se integró el versionado de código, el uso de ramas de trabajo y la automatización del proceso de entrega con un seguimiento de las actividades técnicas relacionadas con la implementación desacoplada del seguimiento general del proyecto, permitiendo la traza de las iteraciones dentro del ciclo de implementación de

cada historia de usuario en GitLab manteniendo un estado transparente de cara al cliente.

- Herramienta de predicción de errores: la integración de la aplicación SonarQube permitió obtener indicadores de calidad del código a partir del análisis de código ante cada confirmación como parte del proceso de integración continua y de las actividades de la gestión de la configuración del software alcanzadas. Estos indicadores se tuvieron en cuenta para iniciar automáticamente el proceso de entrega de los requerimientos involucrados. En el caso de que los hallazgos no permitan obtener un indicador de calidad aceptable, se genera un registro en la traza y las historias de usuario permanecen en estado de finalización pendiente en la gestión del proyecto. Los parámetros de estos indicadores se pueden configurar para adaptarse a las necesidades del proyecto.
- Trazabilidad granular: elementos de la traza que responden a actividades técnicas con mayor relevancia para el equipo de desarrollo (comentarios en la discusión del requerimiento, estimación de esfuerzo, resultados del aseguramiento de calidad y de los procesos de integración continua y de entrega continua).

La funcionalidad del agente de integración implementado para adaptar los mecanismos de sincronización de las herramientas está representado por las líneas punteadas que abarcan a las actividades de Control de Calidad (Gestión de proyecto), Cierre (Gestión de Configuración del Software) y Análisis de código (Predicción de errores). Este agente es quien se encarga de recibir las actualizaciones de todas las actividades e invocar los contratos inteligentes.

En la Figura 2 (izquierda) se muestra el listado de requerimientos agregados a la traza registrada en la red blockchain implementada con Hyperledger Fabric. Esta aplicación centraliza los datos que se originan y registran en las demás herramientas posibilitando que tanto el cliente como cualquier parte interesada pueda acceder de forma más simple a una mayor cantidad de información, presentada además de manera resumida.

Cada ítem de la traza cuenta con un detalle en el que se puede visualizar información adicional como la fecha y hora del registro, la herramienta que lo originó, el enlace al artefacto (si aplica) y el usuario responsable (en caso de haber sido una acción manual). Un ejemplo de esta visualización se ofrece en la Figura 2 (derecha).

Además de la adaptación de los modelos de datos y de contemplar los mecanismos de sincronización específicos de cada herramienta, se debió tener en cuenta la forma de generar los enlaces a los artefactos, ya que esta información no es proporcionada expresamente por los servicios al emitir un evento. De esta manera, se incrementa el nivel de acoplamiento y dependencia ya que se requieren configuraciones específicas y conocer el esquema de URL y cómo fabricar los enlaces a cada servicio, además de la propia configuración de cada servicio para interactuar con el agente.



Figura2. Visualización de los requerimientos en la aplicación de trazabilidad (izq.) y detalles de una transacción (der.)

La situación particular resuelta consistió en configurar Pivotal Tracker para invocar al agente ante actualizaciones en las historias de usuario y, por otra parte, implementar en el agente una consulta activa a SonarQube para obtener datos de los análisis bajo demanda. En el caso de GitLab, la herramienta permite la integración en ambos sentidos mediante webhooks, ejecución de código en el pipeline de integración continua y, también, con acceso por API a los recursos.

Los contratos inteligentes son siempre invocados por eventos externos, por lo que es el agente el que registra la traza. Es importante remarcar que para facilitar esta actividad y garantizar la consistencia de la traza, se optó por evitar la comunicación directa entre los servicios, ya que podría dejar un artefacto en un estado y haber fallado el registro del evento. Se debe velar por la atomicidad entre la traza y la ejecución de la actividad.

5. Conclusiones y trabajo futuro

Mediante la implementación del framework de Calidad 4.0 en el contexto de un proceso de Ingeniería de Software, los desarrolladores y el cliente obtuvieron mayor visibilidad de las expectativas de calidad, fomentando la responsabilidad y la

apropiación del producto. El marco proporcionó un mecanismo para realizar un mejor seguimiento del progreso hacia los objetivos y realizar ajustes informados de manera anticipada, con respecto a otros casos testigo. La integración de métricas personalizadas y KPI se adaptó al flujo de trabajo de integración y entrega continua, convirtiéndose en una parte intrínseca de las revisiones de código, las pruebas automatizadas y otros controles de calidad.

Por otra parte, la transparencia en todo el proceso y la disponibilidad de información de traza de manera simple, centralizada e inmediata permitió potenciar la integración del cliente en todo el proceso de desarrollo, incluyéndolo en más etapas y con una frecuencia de comunicación mayor.

La relación entre el software y la Calidad 4.0 es simbiótica y cíclica. El software actúa como un habilitador de la Calidad 4.0 al proporcionar la infraestructura y las herramientas para implementar la toma de decisiones basada en datos y el monitoreo en tiempo real. Al mismo tiempo, muchas herramientas y técnicas de la industria del software se alinean con el concepto de Calidad 4.0, brindando una visión integrada que impulsa mejores prácticas en los procesos de desarrollo, tales como trazabilidad de extremo a extremo de los requerimientos, el enfoque centrado en el cliente, la mejora continua de los procesos, la detección y el tratamiento temprano de errores, etc. Este ciclo refuerza la adopción de los principios de la Calidad 4.0, lo que da como resultado un ciclo de retroalimentación positiva que beneficia tanto a la industria del software como a la implementación de la Calidad 4.0.

A futuro se propone incluir otros principios, como la Inteligencia Artificial, utilizando herramientas como la asistencia para escritura de código o la predicción de errores; también se propone incluir pruebas de integración y de ejecución de extremo a extremo en la implementación propuesta.

Referencias

1. A review of quality 4.0: definitions, features, technologies, applications, and challenges. Total Quality Management & Business Excellence (2021). doi:
2. 10.1080/14783363.2021.1944082
3. Saha, P., Talapatra, S., Belal, H., y Jackson, V.: Unleashing the Potential of the TQM and Industry 4.0 to Achieve Sustainability Performance in the Context of a Developing Country. Global Journal of Flexible Systems Management 23(4), 495-513 (2022). doi: 10.1007/s40171-022-00316-x
4. Software quality standards. Vojnotehni˘cki Glasnik (2016). doi: 10.5937/
5. VOJTEHG65-10668
6. Riungu-Kalliosaari, L., Mäkinen, S., Lwakatare, L.E., Tiihonen, J., y M¨annist¨o, T.: DevOps adoption benefits and challenges in practice: A case study. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 10027 LNCS, 590 – 597
7. (2016). doi: 10.1007/978-3-319-49094-6_44
8. Villegas Forero, D., y Sisodia, R.: Quality 4.0 -How to Handle Quality in the Industry 4.0 Revolution. Tesis doct., (2020).
9. Agile Software Development: Methodologies and Trends. International Journal of Interactive Mobile Technologies (ijim) (2020). doi: 10.3991/IJIM.
10. V14I11.13269

11. Software Development Process Models Comparison And Assessment Of Degree Of Agility Based On Agile Practices And Performance Implementation On XP And Scrum. International Journal of Scientific & Technology Research 8(12), 1008-1016 (2019)
12. PROCEEDINGS
13. Demi, S., Sánchez-Gordón, M., y Colomo-Palacios, R.: A Blockchain-Enabled Framework for Requirements Traceability. En: Yilmaz, M., Clarke, P., Messnarz, R., y Reiner, M. (eds.) Systems, Software and Services Process Improvement, págs. 3-13. Springer International Publishing, Cham (2021)

Coordinadores:
- Patricia Pesado (UNLP)
- Elsa Estevez (UNS)
- Alejandra Cechich (UNComa)

CACiC 2023

CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

XX Workshop de Ingeniería de Software

> Short Papers

WIS

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Refinamiento del método de análisis de factibilidad del despliegue de sistemas de software: Estudio de caso

Leandro Moreno[1] [0000-0001-7332-3597], Marisa Panizzi[1,2][0000-0002-9699-1544], Rodolfo Bertone[3] [0000-0003-0609-0310]

[1] Programa de Maestría en Ingeniería en Sistemas de Información. Escuela de Posgrado. Universidad Tecnológica Nacional. Universidad Tecnológica Nacional. Facultad Regional Buenos Aires. Medrano 951, C.A.B.A, Argentina.

[2] Departamento de Ingeniería en Sistemas de Información. Universidad Tecnológica Nacional. Facultad Regional Buenos Aires. Medrano 951, C.A.B.A, Argentina.

[3] Instituto de Investigación en Informática (III-LIDI). Facultad de Informática. Universidad Nacional de La Plata. Calle 50 y 120 - La Plata - Bs. As. – Argentina.

leandro.moreno@gmail.com, marisa.panizzi@outlook.com, pbertone@lidi.unlp.edu.ar

Resumen: El despliegue de sistemas de software es el proceso por el cual se hace la transferencia del sistema de software a la empresa cliente y constituye una de las actividades fundamentales dentro del ciclo de vida del software porque el cliente pueda beneficiarse económicamente de su uso. El impacto ocasionado por despliegues fallidos dentro de las organizaciones frecuentemente ocasiona pérdidas de recursos económicos y trae aparejado otras consecuencias negativas en la relación con los clientes. En este trabajo se presentan los resultados de un estudio de caso realizado para examinar la viabilidad de la aplicación de la versión preliminar del método para analizar la factibilidad del despliegue de un sistema de software en un entorno real, con el propósito de refinarlo y completarlo. El estudio se desarrolló en una Pequeña y Mediana Empresa (PyME) de desarrollo de software de Argentina respecto al despliegue de un sistema de autenticación de usuarios para una entidad comercial de dicho país.

Palabras clave: despliegue de sistemas de software, análisis de factibilidad, método, estudio de caso.

1. Introducción

El despliegue de sistemas de software en particular es un proceso crucial del ciclo de vida de desarrollo de software en el que el sistema finalmente estará operativo para que el cliente pueda beneficiarse económicamente de su uso [1]. La implementación de estándares existentes en el mercado frecuentemente se percibe costosa y compleja de realizar [2] por lo que las organizaciones más pequeñas deben enfrentar el desafío de hallar el equilibrio entre sus prácticas cotidianas horizontales y los procedimientos formales, definidos y documentados inherentes a la aplicación de procesos de mejora de software más avanzados [3][4]. Dado que aproximadamente el 80% de las organizaciones dedicadas al desarrollo de software en la Argentina pertenecen a este segmento, las contribuciones realizadas en la materia resultan relevantes [5].

Antes de comenzar con la construcción de un método para el análisis de factibilidad del despliegue de sistemas de software, se realizó un mapeo sistemático de la literatura (en inglés, *systematic mapping study* o SMS) para la construcción del estado del arte e

indagar sobre los aportes existentes sobre el proceso de despliegue de sistemas de software, a través del análisis y observación de las dimensiones "Producto", "Proceso", "Organización", "Prácticas, Herramientas y Técnicas" y "Método" [6]. Sus hallazgos permitieron evidenciar la relevancia de dichas dimensiones y la focalización de los trabajos en el área sobre las primeras dos por sobre las restantes.

En este trabajo se presentan los resultados de un estudio de caso realizado para examinar la viabilidad de la aplicación de la versión preliminar del método de análisis de factibilidad del despliegue de un sistema de software en un entorno real, con el propósito de refinarlo y completarlo.

Este artículo se organiza de la siguiente manera: en la sección 2 se realiza la descripción general de la versión preliminar del método. En la sección 3, se describen las características más significativas del estudio de caso realizado. En la sección 4, se presentan las amenazas que pudieran afectar la validez, y finalmente se presentan las conclusiones y trabajos futuros en la sección 5.

2. Descripción general de la versión preliminar del método

Para el diseño del método para analizar la factibilidad del despliegue de un sistema de software, se tuvieron en cuenta las dimensiones de análisis de los procesos de software propuestas por Petersen *et al.* [6]: "Producto", "Proceso", "Organización", "Prácticas, Herramientas y Técnicas" y "Método".

El método propuesto se basa en el principio de los sistemas de inferencia difuso [7][8], considerando un conjunto limitado de valores lingüísticos para cada respuesta a preguntas contenidas en una plantilla. Cada una de las dimensiones seleccionadas se componen de categorías en las que se agrupan las distintas preguntas, cuya redacción se realiza en lenguaje claro y permite determinar menores o mayores de niveles de certeza o completitud. A cada respuesta se le corresponde un intervalo difuso consistente en cuatro valores numéricos entre cero y diez que representan los puntos de ruptura de su función de pertenencia, un peso relativo y un umbral o valor mínimo esperado, cuyo cumplimiento determina el resultado exitoso de la evaluación. Posteriormente, se obtiene el valor resultante por cada dimensión, y finalmente, el resultado total del método que consistirá en la igualación o superación de un umbral.

En la Tabla 1, se presentan las dos dimensiones seleccionadas para la versión preliminar del método: "Organización" y "Proceso". Por restricciones de espacio, las plantillas utilizadas para la ejecución del estudio de caso se encuentran detalladas en un anexo [9].

3. Descripción del estudio de caso

3.1 Objetivos, características y preguntas de investigación (PI) del estudio de caso

El estudio de caso es de tipo exploratorio y se desarrolló de acuerdo con los lineamientos propuestos por Runeson *et al.* [13]. Su objetivo consistió en evaluar la

aplicación de la versión preliminar de un método para analizar la factibilidad de un proceso de despliegue de un sistema de software, para el que se formularon las siguientes preguntas de investigación (PIs):

PI1: ¿Es útil y aplicable el uso del método por parte del responsable del despliegue de sistemas de software? Esta pregunta apunta a validar la utilidad del método y su aplicación en un escenario de un despliegue real.

PI2: ¿Es necesario refinar y/o completar el método para adaptarlo a las necesidades del contexto industrial? Se busca indagar sobre posibles mejoras a las categorías analizadas y sobre la relevancia, claridad y aplicabilidad del método.

Por otra parte, respecto a las características del estudio de caso es un estudio de caso de tipo holístico y de caso único [14], sustentado por los elementos que se presentan en la Tabla 2.

Tabla 1. Dimensiones, categorías y su descripción.

Dimensión	Categoría	Descripción
Organización (OR)	Estructura organizacional	Adaptabilidad y flexibilidad de la organización para la disposición de sus recursos humanos [10].
	Gestión del conocimiento y comunicación	Administración de los recursos intelectuales de la organización y flujo de información [10].
	Recursos humanos y roles	Generación de roles y perfiles adecuados en cada actividad [10].
	Gestión del proyecto	Institucionalización de procesos, traducido en la adherencia de las prácticas previstas para su desarrollo [10].
	Estándares y regulación	Restricciones y regulaciones gubernamentales que afectan el desempeño de las actividades [11].
Proceso (PR)	Institucionalización	Adherencia, internalización, madurez y proyección organizacional del proceso [12].
	Sistematicidad	Integridad y satisfacción de requisitos necesarios para cumplir con el proceso de despliegue [12].
	Comunicación	Publicidad del proceso de despliegue con los interesados [12].
	Riesgo	Acciones de identificación y mitigación de riesgos asociados a la ejecución de cada paso previsto para el despliegue [12].

Tabla 2. Contexto, caso y unidad de análisis.

Contexto	Caso	Unidad de análisis
PyME desarrolladora de software de Argentina de 110 empleados.	Despliegue de versión incremental para sistema de autenticación de una entidad comercial.	Versión preliminar del método para analizar la factibilidad del despliegue de un sistema.

3.2 Recolección de datos, análisis e interpretación de resultados

De acuerdo con la división de las técnicas de recolección de datos [13], se realiza una recolección de tercer grado. En primer lugar, se realizó una capacitación virtual de dos horas al usuario responsable del despliegue del sistema a través de *Microsoft Teams*. Luego, se le brindó una plantilla para ser completada en la jornada de planificación de despliegue previsto. A continuación, en la Tabla 3, se resumen los resultados obtenidos y posteriormente, las respuestas a las preguntas de investigación (PIs):

Tabla 3. Resumen de hallazgos.

Dimensión	Cantidad de preguntas	Respues-tas	Tiempo (en minutos)	Observación		
				Relevancia	Precisión	Claridad
Organización	30	30	7	2	3	1
Proceso	30	30	5	2	1	1

PI1: ¿Es útil y aplicable el uso del método por parte del responsable del despliegue de sistemas de software?

Se determinó que el método es útil y aplicable, al lograr completarse en su totalidad en 12 minutos, recibiendo observaciones sobre su pertinencia solamente en 4 preguntas.

PI2: ¿Es necesario refinar y/o completar el método para adaptarlo a las necesidades del contexto industrial?

Las observaciones recibidas sobre las preguntas del método pueden agruparse de la siguiente manera:

- Relevancia: de un total de 60 preguntas propuestas, a 4 preguntas (6,7%) se le cuestionó la importancia de las variables consultadas sobre proceso de despliegue.
- Precisión: 4 preguntas (6,7%), recibieron una observación sobre la precisión de la respuesta posible al analizar el comportamiento de los miembros del equipo.
- Claridad: respecto a esta variable, 2 preguntas (3,35%) fueron identificadas con alguna clase de falencia en torno a la comprensión de la presentada.

4. Amenazas a la validez

Para analizar la validez del estudio, se tuvieron en cuenta los factores propuestos por Runeson *et al.* [13]:

- Validez de constructo. El método se validó a través análisis de los datos recopilados de un caso real que permitió responder a las PIs definidas.
- Validez interna. Se reconoce la necesidad de combinar la documentación generada en el método con otro tipo de fuentes para mayor precisión y validez del método.
- Validez externa. El uso de un solo estudio de caso puede limitar la generalización de los resultados. En este caso se considera necesario informar sobre estos hallazgos, ya que sirve como un incentivo para repetirlo con otros estudios de caso.
- Fiabilidad. Aunque los datos han sido analizados entre los coautores, fueron recopilados por un solo autor, pudiendo ser una amenaza para la investigación.

5. Conclusiones y trabajos futuros

Los resultados sobre el análisis de la versión preliminar del método permiten concluir que el método es el útil y aplicable. Por otra parte, se han recogido observaciones sobre la pertinencia, precisión y completitud de las preguntas, sobre las que en un próximo paso se refinarán las preguntas con algún tipo de déficit en estos aspectos.

Dentro de las líneas de trabajo futuro se encuentran: (a) avanzar en el análisis de las dimensiones "Producto" y "Técnicas, Herramientas y Prácticas" para completar una nueva versión del método; b) continuar con la validación del método mediante otros estudios de casos en el contexto industrial.

Referencias

1. Panizzi, M. Genero M. and R. Bertone, R. Refining a Software System Deployment Process Model through Empirical Studies. DOI: 10.24215/16666038.23.e06 (2023).
2. Jansen S., Brinkkemper S. Definition and validation of the key process of release, delivery and deployment for product software vendors: Turning the ugly duckling into a swan IEEE International Conference on Software Maintenance, art. 4021334, pp. 166-175 (2006).
3. Subramanian N. The software deployment process and automation. pp. 28-34 (2017).
4. Paredes I., Carvalho J. Research in Progress: Understanding the process of implantation IT Enterprise Applications in Small and Medium Enterprises (SMEs). DOI:10.18803/capsi.v17.270-283 (2017).
5. OPPSI-CESSI. Reporte anual del sector de software y servicios informáticos de la República Argentina del año 2018. Cámara de la Industria Argentina de Software (2018).
6. Petersen, K. Wohlin, C.: Context in industrial software engineering research (2009).
7. Jang, J.S.R. (1993) ANFIS: Adaptive-Network-Based Fuzzy Inference System (1993).
8. Pythel, P. Viabilidad y Estimación de proyectos de explotación de información. Tesis doctoral. Facultad de Informática. Universidad Nacional de La Plata (2014).
9. Moreno L., Panizzi M., Bertone R. Anexo. Refinamiento del método de análisis de factibilidad del despliegue de sistemas de software: Estudio de caso. Disponible en: https://doi.org/10.6084/m9.figshare.23833812
10. Nagappan, N., Murphy, B., Basili, V. The influence of organizational structure on software quality. DOI :10.1145/1368088.1368160 (2018).
11. Project Management Institute. A guide to the Project Management Body of Knowledge (6th ed.) (2017).
12. CMMI Product Team, "CMMI for Development, Version 1.3," Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania (2010).
13. Runeson P., Höst M., Rainer A., Regnell B. Case study research in software engineering: guidelines and examples. Wiley Publishing, Hoboken (2012).
14. Yin, R.K. Case study research: design and methods. 5th Edition. Sage Publications (2014).

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Encuesta sobre el uso y la relación entre Escenarios y User Story Mapping

Andrea Alegretti[1] [0009-0003-1021-6921], Leandro Antonelli[2] [0000-0003-1388-0337], Marisa Panizzi[3] [0000-0002-9699-1544]

[1]Programa de Maestría en Ingeniería de Software, Facultad de Informática, Universidad Nacional de La Plata, Calle 50 y 120 - La Plata - Bs. As. – Argentina.
[2]Lifia, Facultad. de Informática, Universidad Nacional de La Plata, Calle 50 y 120 - La Plata - Bs. As. – Argentina.
[3]Instituto de Tecnología e Ingeniería, Universidad Nacional de Hurlingham, Av. Vergara 2222 (B1688GEZ) - Villa Tesei - Bs. As. Argentina
andyalegretti@gmail.com, lanto@lifia.info.unlp.edu.ar, marisa.panizzi@unahur.edu.ar

Resumen. La ingeniería de requerimientos tiene por objetivo brindar técnicas y herramientas con el fin de elicitar, analizar, documentar y validar los requerimientos que permitirán definir el producto de software a desarrollar. Sin embargo, dado que el software es hoy en día una herramienta clave en los negocios, es necesario alinear los requerimientos con las necesidades de los clientes y la relación con los objetivos del negocio. Existen varias técnicas que permiten hacer esta alineación, sin embargo, son generalmente técnicas clásicas que requieren de mucho esfuerzo, en este sentido, es necesario contar con una técnica ágil. Este artículo describe un estudio exploratorio basado en una encuesta para evidenciar si los profesionales de la industria del software tienen conocimiento y utilizan dos herramientas representativas de la etapa de exploración del producto como Escenarios y *User Story Mapping*, y que además existe una relación entre ellas. En esta encuesta participaron 112 profesionales de la industria del software de Argentina. Los resultados reportan que la mayoría definió alguna vez Escenarios utilizando el modelo de Leite y además conocen la herramienta de *User Story Mapping*. Dentro de los hallazgos más significativos podemos evidenciar que los profesionales de la industria del software consideran que puede existir una relación entre las actividades principales que se definen en el *User Story Mapping* y la identificación de Escenarios.

Palabras claves: Escenarios, user story mapping, requerimientos, encuesta.

1. Introducción

Uno de los principales factores de fracaso de los proyectos de software se debe a la carencia de un proceso adecuado de definición y entendimiento del problema y a la definición poco clara de las necesidades del cliente [1], como así también al descubrimiento de herramientas acordes que permitan gestionar los requerimientos, sus cambios a lo largo del proyecto y la alineación entre los requerimientos con las necesidades de los clientes y la relación con los objetivos del negocio.

Los requisitos son, por definición "una condición o necesidad de un usuario para resolver un problema o alcanzar un objetivo" [2] y, en la línea de [3] "expresan las necesidades y limitaciones impuestas a un producto de software que contribuyen a la solución de algún problema en el mundo real".

En la búsqueda por mejorar el proceso de gestión de requisitos mediante una técnica ágil, este trabajo se centró en analizar y encontrar una relación entre dos herramientas claves que se utilizan en las actividades que se realizan en esta etapa: Escenarios y *User Story Mapping*.

Un escenario es un artefacto que describe situaciones en un dominio usando lenguaje natural en donde ocurre una situación específica que surge en un determinado contexto con el fin de lograr algún objetivo [4]. Hay un conjunto de pasos para alcanzar ese objetivo (episodios). En los episodios, los agentes activos (actores) utilizan materiales, herramientas y datos (recursos) para realizar alguna acción específica. Mientras que dentro del marco de trabajo SCRUM en la finalización de la etapa de agile inception, existe una herramienta visual denominada *User Story Mapping* que fue creada por Jeff Patton [5] y permite generar una representación visual de todas las funcionalidades que componen un sistema reflejando lo que realiza el mismo. Los elementos que lo conforman son: Actividades, Tareas y Subtareas (Historias de Usuario).

Por todos los inconvenientes enunciados anteriormente y con el propósito de incrementar la calidad y capacidad del proceso de gestión de requerimientos, se decidió como objetivo de investigación a largo plazo, explorar el diseño de un método que permita especificar, mejorar y mantener Escenarios mediante la herramienta de *User Story Mapping.*

Con el propósito de recolectar evidencia sobre el uso de estas herramientas en la industria del software y la relación entre ellas se decidió realizar un estudio exploratorio a través de una encuesta realizada según las directrices de Molléri *et al*. [6].

Este artículo se estructura de la siguiente manera: en la Sección 2 se describe la planificación de la encuesta, en la Sección 3 se describe su ejecución. Los resultados se presentan en la Sección 4. En la Sección 5 se exponen las conclusiones y trabajos futuros.

2. Planificación de la encuesta

Objetivos y preguntas de investigación. El uso de la plantilla de GQM (*Goal-Question-Metric*) [7] nos permitió formular el objetivo de la encuesta de la siguiente manera: "***Analizar*** la utilización de las herramientas de Escenarios y *User Story Mapping* ***con el propósito de*** conocer la utilización de las mismas en el proceso de gestión de los requerimientos en el desarrollo de software ***con respecto a*** descubrir la relación entre ellas ***desde el punto de vista*** de profesionales de la industria del software".

Las preguntas de investigación (PI) que guiaron esta encuesta son las siguientes:

PI1: ¿Qué características tienen las empresas en relación con la gestión de requerimientos?

PI2: ¿Qué conocimiento tiene la industria del software sobre la herramienta User Story Mapping?

PI3: ¿Qué conocimiento tiene la industria del software sobre la herramienta Escenarios?

PI4: ¿Los profesionales de la industria del software consideran que existe alguna relación entre las herramientas de User Story Mapping y Escenarios?

Proceso de ejecución. Este proceso consta de: 1) Diseño de un formulario para llevar un registro sistemático de la ejecución de la encuesta. Este se compone de dos partes: a) Proceso de envío de la encuesta: cantidad de envíos y a quienes. y b) Proceso de seguimiento del envío: mails erróneos, cantidad de cuestionarios respondidos y fecha de respuesta, 2) Envío de la encuesta por correo electrónico. Se diseñó un texto de presentación en el cual se menciona el propósito de la investigación, quiénes participan, agradecimiento por la colaboración y una invitación a que el encuestado difunda la encuesta entre sus contactos, 3) Revisión diaria de encuestas respondidas, 4) Extracción de las respuestas y 5) Extracción de los datos.

Población. La población a la cual se decidió enviar la encuesta son profesionales de la industria del software, incluyendo también estudiantes de carreras de grado y de Posgrado, investigadores y docentes de asignaturas pertinentes a la Ingeniería y gestión de requisitos.

Se utilizaron diferentes estrategias para seleccionar la muestra, una de ellas consistió en publicar la encuesta en grupos de WhatsApp y Telegram, otra estrategia que se utilizó fue la de publicar la encuesta en LinkedIn y por último se realizó el envío de la encuesta de manera directa por parte de los autores.

Diseño de la encuesta. Se diseñó un cuestionario autoadministrado a los encuestados y se utilizó la herramienta Google Forms.

Para el diseño del cuestionario se definieron 2 preguntas para contextualizar a los participantes y 5 preguntas para conocer el uso de las dos herramientas que se proponen (Escenarios y User Story Mapping) y la relación entre ellas. El cuestionario y la correspondencia entre las preguntas del cuestionario con las PI se encuentran en un apéndice por restricciones de espacio [8]. El cuestionario completo se encuentra disponible en https://forms.gle/VFSfUxUJgZf6m1ua6

Validación. Se realizó una prueba piloto con un grupo reducido de 10 participantes que permitió verificar el tiempo de respuesta inicial estimado de 5 minutos, quedando reducido a 3 minutos. También se logró validar la terminología empleada, los participantes de la prueba piloto confirmaron su claridad y legibilidad.

3. Ejecución de la encuesta

Reclutamiento de los participantes. Se utilizaron grupos de *WhatsApp* y de *Telegram* dónde sus miembros son profesionales del ámbito de sistemas, docentes, investigadores, alumnos y ex alumnos de carreras de grado y posgrado de informática.

Además, de manera directa se envió a contactos de profesionales de la industria del software y ex alumnos de carreras de grado y posgrado de informática de uno de los investigadores.

Gestión de las respuestas. La gestión de la ejecución de la encuesta se realizó de acuerdo con el procedimiento definido en la sección 2.

Análisis de los datos. Se analizó la veracidad de todas las direcciones de correo electrónico que respondieron la encuesta, además se analizaron las respuestas que se

obtuvieron de manera cuantitativa y se plantearon relaciones entre las preguntas que contaban con mayor porcentaje de respuestas positivas. Finalmente, se analizó una pregunta en particular que indica la relación directa que existe entre los Escenarios y *User Story Mapping*.

4. Análisis e interpretación de los resultados obtenidos

En esta sección se presentan las respuestas a las PIs:

PI1: ¿Qué características tienen las empresas en relación con la gestión de requerimientos?

La mayoría de los participantes son Desarrolladores (29,46 %), en segundo lugar los Líderes de Proyecto (25,89 %), en tercer lugar, roles ágiles (16,96 %), el cuarto lugar los Analistas de BI (10.71%), en quinto lugar los Analistas Funcionales (9,82%) y por último Director (7,14%).

El 81,25% de los participantes trabajan en empresas privadas y cuentan con herramientas ágiles que les permiten gestionar los requerimientos, mientras que el 18,75% trabaja en empresas públicas y tienen conocimiento de las dos herramientas.

PI2: ¿Qué conocimiento tiene la industria del software sobre la herramienta User Story Mapping?

El 75% de los participantes indicaron que conocen la herramienta y que les ayuda a diseñar y comprender el flujo de trabajo a través del tiempo, describiendo los procesos de negocios de cada uno de los actores que forman parte del sistema, a través de las tareas que deben llevarse a cabo para completar cada uno de estos procesos.

Además, el 64,3% de los encuestados indicaron que les resulta sencilla la técnica debido a que, en este mapa, los equipos crean un esquema dinámico de las interacciones de un usuario representativo con el producto, evalúan qué pasos tienen el mayor beneficio para el usuario y priorizan lo que se debe construir.

También solo el 28,6% de los participantes indicaron que están utilizando la herramienta de *User Story Mapping* en sus proyectos de desarrollo de software para representar gráficamente los procesos que componen el mismo, los actores y los requerimientos que solicitan.

PI3: ¿Qué conocimiento tiene la industria del software sobre la herramienta Escenarios?

El 50,9% definió alguna vez en su experiencia laboral y/o universitaria escenarios utilizando el modelo de Leite [9] que consta de 6 atributos (i) un título; (ii) una meta o objetivo a alcanzar; (iii) un contexto; (iv) los recursos; (v) los actores y (vi) el conjunto de episodios.

PI4: ¿Los profesionales de la industria del software consideran que existe alguna relación entre las herramientas de User Story Mapping y Escenarios?

La mayoría de los participantes de la encuesta (96,4%) creen que existe una relación directa entre lo que es la identificación y definición de Escenarios y cada una de las Actividades principales (backbone) que componen un *User Story Mapping*.

5. Conclusiones y trabajos futuros

En base a los hallazgos de la encuesta podemos concluir que:

- La mayoría de los encuestados conocen las herramientas de Escenarios y *User Story Mapping* y trabajaron en algún momento en sus proyectos laborales y/o académicos con ambas herramientas.
- A la mayoría de los encuestados les resulta sencillo aplicar la herramienta de *User Story Mapping* para definir sus productos de software, debido a que los equipos crean un esquema dinámico de las interacciones de un usuario representativo con el producto, evalúan qué pasos tienen el mayor beneficio para el usuario y priorizan lo que se debe construir.
- La mayoría de los encuestados definió en algún momento Escenarios según el modelo planteado de 6 atributos de Leite para describir una situación del dominio y un conjunto de acciones a realizar a partir de él para alcanzar una meta.
- Se logró evidenciar una relación directa entre cada una de las actividades principales que componen un *User Story Mapping* y la definición e identificación de Escenarios. Es decir que se podrían identificar tantos Escenarios como actividades involucradas dentro del mapa.

A partir de los resultados obtenidos en la encuesta, se explorará el diseño de un método que permita definir Escenarios mediante la utilización de *User Story Mapping* y se ajuste a las características que propone Leite.

Antes de comenzar con el diseño del método, se realizará un mapeo sistemático de la literatura.

Referencias

1. L. F. Londoño, R. Anaya y M. Silva, «Análisis de la ingeniería de requisitos orientada por aspectos según la industria del software,» Revista EIA, nº 9, pp. 43-52, (2008).
2. IEEE, «IEEE Software Engineering Standard: Glossary of Software Engineering Terminology,» (1993). [En línea]. Available: http://dis.unal.edu.co/~icasta/GGP/_Ver_2012_1/Documentos/Normas/610-12-1990.pdf.
3. IEEE, SWEBOK Guide V3.0, Piscataway: IEEE, (2014).
4. Carrol, J. M.: "Five reasons for scenario-based design," in Proceedings of the 32nd Annual Hawaii International Conference on Systems Sciences, (1999).
5. Patton,J: "User Story Mapping: Discover the Whole Story, Build the Right Product", (2014)
6. Molléri J.S., K. Petersen K., E. Mendes E. An empirically evaluated checklist for surveys in software engineering. Information and Software Technology 119 (2020) 106240 (2020).
7. Basili V., y Rombach D. The TAME Project: Towards Improvement-Oriented Software Environments. IEEE Transactions on Software Engineering, 14(6) (1988).
8. Panizzi M. Apéndice-Encuesta sobre el uso y la relación entre Escenarios y User Story Mapping. Disponible en: https://doi.org/10.6084/m9.figshare.23908509 (2023).
9. Leite, J. C. S. d. P., Rossi, G., Balaguer, F., Maiorana, V., Kaplan, G., Hadad, G. and
10. Oliveros, A.:"Enhancing a requirements baseline with scenarios," Requirements Engineering, vol. 2, no. 4, pp. 184-198, (1997).

Coordinadores:

- Pablo Fillottrani (UNS)
- Marcelo Estayno (UNSAM)
- Alicia Mon (ITBA)

CACiC 2023

CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

XVI Workshop de Innovación en Sistemas de Software

> Full Papers

WISS

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Bot de Trading.
Control movimientos de Cripto Monedas.

Juan Pablo Pollini[1], Facundo Ferro[2], Marcelo Taruschio[3], Rodolfo Bertone[4]

[1] Alumno de Grado, FACEI UCALP
[2] Profesor Lic. en Sistemas, FACEI UCALP
[3] Director de Carrera, Profesor, FACEI UCALP
[4] Profesor y Secretario Académico a Cargo, FACEI UCALP
jppollinijp@gmail.com, facundoferro@gmail.com,
marcelo.taruschio@ucalp.edu.ar, rodolfo.bertone@ucalp.edu.ar

Abstract. En el mundo de las inversiones, existen diferentes tipos de activos que permiten a los inversores destinar su capital o ahorros con el objetivo de generar ganancias. Estas ganancias están sujetas a las fluctuaciones de precios del activo en el que se invierte. En la actualidad, uno de los activos más novedosos lo constituyen las criptomonedas

Las criptomonedas han experimentado un crecimiento exponencial en los últimos años, atrayendo la atención de inversores de todo el mundo. Estas monedas digitales funcionan con tecnología blockchain, que garantiza la seguridad y la transparencia de las transacciones realizadas.

En este contexto, cada vez más inversores están interesados en aprovechar las oportunidades que ofrecen las criptomonedas para obtener ganancias significativas. Sin embargo, la inversión en criptomonedas puede ser compleja y requiere un análisis cuidadoso de las diferentes variables para la inversión.

Este trabajo, se basa en un enfoque específico para invertir en criptomonedas: el uso de sistemas automatizados desarrollados utilizando lenguajes de programación. Estos sistemas se basan en algoritmos y estrategias predefinidas que permiten tomar decisiones de compra y venta de forma automatizada, teniendo en cuenta varios parámetros como el precio, el volumen de negociación y las tendencias del mercado.

Keywords: Criptomonedas, Algoritmos, Toma de decisiones.

1. Introducción

En los últimos años, las criptomonedas han captado la atención tanto de la comunidad financiera como del público en general. Estas monedas digitales descentralizadas, como el Bitcoin, Ethereum y muchas otras, han revolucionado la forma en que se concibe el dinero y las transacciones financieras. Al utilizar tecnología blockchain, una base de datos distribuida y segura, las criptomonedas ofrecen un enfoque novedoso para el intercambio de valor y han abierto nuevas posibilidades en el mundo de las inversiones.[1][2]

El auge de las criptomonedas ha llevado a un creciente interés por parte de los inversores en este emergente mercado financiero. A diferencia de las inversiones tradicionales en acciones, bonos o inmuebles, las criptomonedas ofrecen una serie de características únicas que las distinguen. En primer lugar, su naturaleza descentralizada

elimina la necesidad de intermediarios. Además, la tecnología blockchain garantiza la seguridad y la transparencia de las transacciones, lo que ha generado.[3]

La volatilidad de las criptomonedas es otro aspecto que ha llamado la atención de los inversores. Los precios de las criptomonedas pueden experimentar cambios significativos en períodos cortos de tiempo. La falta de regulación y la incertidumbre en torno al panorama legal de las criptomonedas han sido factores que han contribuido a su volatilidad, pero también han planteado desafíos y preocupaciones para los inversores.[4]

A medida que el mercado de las criptomonedas ha evolucionado, han surgido diferentes enfoques de inversión. Algunos inversores optan por mantener criptomonedas a largo plazo, considerándolas como una reserva de valor similar al oro. Otros buscan oportunidades de trading a corto plazo, aprovechando la volatilidad del mercado para obtener beneficios. Además, han surgido nuevas formas de inversión, como los fondos de inversión en criptomonedas y las ofertas iniciales de moneda, que permiten a los inversores participar en proyectos relacionados con criptomonedas y tecnología blockchain.

A medida que las criptomonedas continúan ganando terreno en el panorama financiero, es importante comprender los fundamentos y los riesgos asociados con este tipo de inversiones. Se han creado nuevas oportunidades para la inversión y la especulación. Además, se han desarrollado herramientas y plataformas específicas para facilitar la inversión en criptomonedas, brindando a los inversores acceso a información en tiempo real, análisis técnico y herramientas de gestión de cartera. Estas herramientas permiten a los inversores tomar decisiones más informadas y aprovechar las oportunidades en este mercado muy dinámico.[6][7]

Aunque el mundo de las criptomonedas ha generado entusiasmo y oportunidades, también presenta desafíos y consideraciones importantes. La seguridad de los activos digitales es una preocupación fundamental. Los inversores deben tomar medidas para proteger sus activos mediante el uso de billeteras seguras y la adopción de buenas prácticas de seguridad cibernética. Además, la volatilidad de los precios de las criptomonedas puede ser un obstáculo significativo para los inversores menos experimentados. La comprensión de los factores que influyen en la volatilidad de los precios, como las noticias del mercado, la regulación gubernamental y los eventos macroeconómicos, es crucial para tomar decisiones y gestionar el riesgo de manera efectiva.[8][9]

1.1 Trading financiero

El trading financiero es la acción de comprar y vender diferentes tipos de activos financieros que cotizan en distintos mercados con el objetivo de obtener una rentabilidad a partir de la especulación del precio que tendrán estos en un momento futuro. En todos los mercados, existe la posibilidad de operar al alza cuando especulamos que el precio va a subir y a la baja cuando creemos que va a bajar. [10]

No existe una certeza de lo que ocurrirá con el precio, por lo tanto, todos los análisis y estudios que se realizan sobre los activos son especulativos y probabilísticos, es decir, tenemos una probabilidad de que algo ocurra y esto puede generar una ganancia, una

pérdida o ninguna de las dos (en este caso, puede ser que únicamente se abone la comisión por operar dependiendo del mercado). [11]

Existe una clasificación básica de plazos, lo cual es relativo dependiendo la fuente consultada dado que no existe un estándar básico sobre estas definiciones. A continuación, se establecen los tiempos que se consideran apropiados para cada uno:

- Corto plazo: operaciones que pueden durar desde segundos a pocos días;
- Mediano plazo: operaciones que pueden durar desde una semana a un mes;
- Largo plazo: operaciones que pueden durar desde un mes a varios años.

Por otro lado, tenemos los principales tipos de trading dependiendo la operatoria llevada a cabo:

- Scalping: las operaciones duran muy poco tiempo (segundos o minutos) con el objetivo de obtener pequeñas ganancias rápidamente;
- Intraday: las operaciones se abren y cierran en el mismo día, tratando de detectar la tendencia principal del día en cuestión (dura de minutos a horas);
- Swing: se busca operar el cambio de tendencia manteniendo las operaciones abiertas varios días, meses o años, por lo tanto, requiere de más paciencia.

El tipo de trading variará de acuerdo a la estrategia de cada trader y los objetivos que este se proponga frente a esta actividad.

2. Caso de Estudio

El trabajo a realizar surge a partir del interés de los autores bajo el concepto de “hacer que el dinero trabaje para las personas”. Esta perspectiva llevó a estudiar los mercados financieros y su comportamiento para lograr ese fin. Invertir no es una tarea sencilla, requiere de un amplio conocimiento, dependiendo del tipo de inversionista a ser.

A partir de ese concepto se planteó la idea de automatizar las operaciones que siempre fueron manuales, con el objetivo de mejorar la actividad. Este enfoque brinda una cierta libertad, ya que no se depende de las fluctuaciones del mercado. Para el caso en que se realiza de forma manual. el tiempo dedicado frente a las pantallas varía en función del tipo de operaciones que se realicen. Es importante tener en cuenta que pasar largas horas frente a ellas puede tener consecuencias negativas para la salud y el bienestar físico.

Además, el aspecto psicológico desempeña un papel importante en las operaciones financieras. En muchas ocasiones, las emociones pueden jugar en contra, llevando a pérdidas significativas en cuestión de segundos o minutos.

En respuesta a todas estas consideraciones, una posible solución es el trading automático, que se propone como una alternativa en este estudio. Al eliminar la influencia de las emociones y la codicia en las operaciones, se puede ganar más tiempo libre y evitar estrés y nerviosismo, entre otros aspectos negativos. Es posible obtener ingresos pasivos a largo plazo de manera sostenida, siempre y cuando la estrategia implementada por el bot se adapte al contexto del mercado.

Es importante destacar que el trading automático no está exento de riesgos y es fundamental realizar un análisis riguroso y una correcta configuración del sistema para

maximizar las oportunidades de éxito. No obstante, representa una opción atractiva para aquellos inversores que buscan optimizar su tiempo y minimizar los impactos negativos asociados a la operativa manual.

2.1 Objetivos

Objetivos del estudio:

1. Desarrollar un sistema automatizado rentable para operar en el mercado de criptomonedas, maximizando el rendimiento de las inversiones realizadas.
2. Ejecutar operaciones de manera eficiente y automática, utilizando algoritmos informáticos para reducir la intervención manual constante.
3. Minimizar la dependencia del operador respecto a las fluctuaciones del mercado, evitando decisiones influenciadas por emociones y sesgos cognitivos.
4. Liberar tiempo para el inversor al delegar las operaciones al sistema automatizado, permitiendo un mejor equilibrio entre trabajo y actividades personales.
5. Preservar la salud física y mental del inversor al reducir el estrés y la presión asociados con el monitoreo constante del mercado.
6. Obtener ganancias consistentes y sostenibles a largo plazo mediante una estrategia de inversión respaldada por el sistema automatizado.
7. Eliminar errores emocionales en las operaciones al basarse en criterios objetivos y análisis técnico proporcionados por el sistema.
8. Implementar una adecuada gestión del capital invertido, con estrategias de gestión de riesgos y diversificación de cartera para proteger el capital.
9. Filtrar el ruido del mercado en temporalidades bajas, enfocándose en análisis más sólidos y tendencias de mediano y largo plazo.
10. Concientizar a los inversores sobre el trading en criptomonedas, proporcionando información relevante sobre beneficios y riesgos, y promoviendo prácticas prudentes en las operaciones financieras.

Estos objetivos guían el desarrollo y análisis del estudio, con el fin de evaluar la viabilidad y efectividad del sistema automatizado en el mercado de criptomonedas. [12]

3. Solución propuesta

Los requerimientos necesarios para implementar el proyecto se dividieron en dos grupos. El primero de ellos se refiere al hardware. Para llevar a cabo el proyecto, es necesario contar con una conexión estable a internet que no tenga latencia ni retardos en el envío de datos. Esto es importante para poder colocar órdenes en el mercado en tiempo real. Como segundo aspecto del hardware, es necesario disponer de suficiente poder de cómputo que permita realizar un análisis correcto del mercado en varios pares de criptomonedas de manera simultánea, lo que permitirá la ejecución de múltiples instancias de la aplicación desarrollada. La elección del hardware dependerá del capital disponible para el proyecto, ya que, para evitar el mantenimiento, podría considerarse contratar el servicio de un tercero, como un VPS donde se pueda alojar el bot.

El segundo grupo está constituido por el software. Para la solución prevista, fue necesario desarrollar dos componentes: el bot de trading, que permite realizar análisis, ejecutar operaciones y administrar el capital de inversión, y el código del bot de Telegram, que se encarga de comunicar los resultados al usuario. Además, se utilizo un "Exchange" desde el cual fue posible habilitar una cuenta de usuario, transferir el capital a invertir y operar en el mercado. Por último, se necesita desarrollar una API que interactúe con el Exchange para permitir el intercambio de los datos e información necesaria y enviar las órdenes que el bot disponga. De esta manera, se logra obtener los datos del mercado y de la cuenta del usuario, analizarlos y ejecutar cada una de las operaciones deseadas.

El objetivo del bot de trading es enviar órdenes de compra en el mercado basándose en el cálculo y análisis de ciertos indicadores técnicos, considerando el precio de la criptomoneda y el período de tiempo en el cual se estudia el mercado. Las operaciones son ejecutadas por el Exchange luego de recibir una orden del bot a través de la API del Exchange. Se pueden ejecutar varias instancias del bot, lo que permite analizar diferentes pares de criptomonedas, comparándolos con la criptomoneda USDT (cuyo valor equivale aproximadamente a un dólar). Por ejemplo, el par BTC/USDT (Bitcoin con respecto al dólar).

El bot tiene limitaciones debido a que solo realiza cálculos de indicadores técnicos de baja complejidad, lo que permite obtener un análisis más rápido del mercado y colocar órdenes con mayor celeridad. Esta limitación está determinada por el tipo de hardware utilizado. Para administrar los indicadores técnicos en las unidades de tiempo definidas, se requiere un gran procesamiento de información. En el proyecto, se utilizaron computadoras clásicas de mercado con procesadores i5 o rayzen 5, por lo que es necesario administrar el uso de indicadores y unidades de tiempo para obtener respuestas adecuadas en el tiempo disponible. Los desarrollos realizados podrían incorporar más indicadores si fuera posible acceder a hardware dedicado con mayor potencia de cómputo. A partir de la solución generada, fue posible analizar la tendencia del mercado y obtener un mayor número de ganancias en el movimiento de las criptomonedas.

El contexto del mercado, donde se analiza el comportamiento de las criptomonedas, es muy cambiante y constantemente se puede tener un análisis diferente dependiendo de la temporalidad. El bot desarrollado no tiene la posibilidad de llevar a cabo un "trailing stop", es decir, mover el stop loss para disminuir las pérdidas y asegurar las ganancias a medida que el precio se mueve a favor. Esta característica será incorporará en futuras versiones del bot de trading.

3.1 Implementación

Para abordar los diversos componentes de software, el proyecto utiliza el lenguaje de programación Python como base. Esta elección se basa en las numerosas ventajas que ofrece Python, como su flexibilidad, facilidad de uso y amplia comunidad de desarrollo. Python proporciona un entorno propicio para el desarrollo de soluciones tecnológicas avanzadas, y su sintaxis clara y legible facilita la escritura de código eficiente y de fácil mantenimiento. Además, cuenta con una amplia gama de bibliotecas y frameworks que brindan funcionalidades adicionales y simplifican el proceso de desarrollo. [13]

En particular, el proyecto aprovecha la incorporación de diversas librerías de Python que resultan fundamentales para la resolución de cálculos complejos y el análisis de resultados presentados a los usuarios. Estas librerías específicas, permitieron optimizar el rendimiento y la precisión de los algoritmos que se implementan.

Para llevar a cabo el proyecto, es necesario ejecutar el software del bot en hardware físico, ya sea en un servidor local o en un VPS (servidor privado virtual). El servidor físico requiere configuración inicial, mantenimiento y reparaciones, así como una conexión estable a internet y suministro eléctrico constante. El hardware necesario no es sofisticado, por lo que se puede utilizar una computadora de escritorio o una notebook como servidor físico, aprovechando los recursos disponibles para el procesamiento de datos.

En el proyecto presentado, el bot se ejecuta en el sistema operativo Ubuntu 20.04.3 LTS, una distribución de software libre de GNU/Linux. Ubuntu es fácil de usar, permite la instalación de programas y aplicaciones, ofrece actualizaciones frecuentes y brinda libertad de uso y distribución. Está basado en Debian y es adecuado tanto para uso de escritorio como para servidores.

3.2 Aplicaciones de terceros utilizadas

Se utilizarán las siguientes aplicaciones de terceros:

- Binance: es el exchange líder en el mercado de criptomonedas y NFTs. Ofrece una amplia gama de productos, como futuros, compra/venta de criptomonedas, tarjeta de débito, stacking, entre otros.
- Nagios: es un proyecto de código abierto iniciado en 1999 y con crecimiento colaborativo dentro de la comunidad que lo creara. Su función principal es monitorear la infraestructura tecnológica para garantizar el correcto funcionamiento de sistemas, aplicaciones, servicios y procesos comerciales. En caso de fallas, Nagios alerta al personal técnico para que pueda tomar medidas correctivas antes de que afecten los procesos comerciales, usuarios o clientes. El proyecto lo utiliza para monitorear los procesos de la aplicación bots y el estado del servidor.
- Telegram: es una aplicación de mensajería en el proyecto, para comunicarle al usuario la información del bot de trading.

3.3 Estrategia

La estrategia de solución empleada se basa en aprovechar las tendencias del instrumento operado, utilizando un indicador técnico conocido como media móvil simple (SMA). Las SMA son medidas en periodos. En este caso, se utilizan tres SMAs con diferentes períodos: cuatro, nueve y dieciocho.

La estrategia se asemeja a un semáforo, asignando un color a cada SMA. La SMA de cuatro períodos se representa en verde, la de nueve períodos en amarillo y la de dieciocho períodos en rojo. Las operaciones del bot se abren o cierran en función de las condiciones que se cumplan en relación a los valores de cada SMA. Cuando la SMA de

cuatro períodos cruza hacia abajo las medias de nueve y dieciocho períodos, se identifica una tendencia bajista. Por otro lado, cuando la SMA de cuatro períodos cruza hacia arriba las medias de nueve y dieciocho períodos, se identifica una tendencia alcista.

Las operaciones que se llevan a cabo con el bot son de compra, el bot adquiere unidades del activo en el mercado spot. Estas unidades se mantienen hasta que el precio actual supere el precio de compra, con el objetivo de obtener beneficios en lugar de sufrir pérdidas. Para realizar una operación de compra, se verifica que la SMA de cuatro períodos sea mayor que las medias de nueve y dieciocho períodos. Esta condición indica una señal positiva para abrir una posición de compra. Asimismo, cuando la SMA de cuatro períodos se iguala a la SMA de nueve períodos y el precio de entrada es menor que el precio actual, se ejecuta la operación de venta, buscando obtener ganancias. Hay que tener en cuenta que en algunos casos puede ser necesario esperar un tiempo prolongado si el precio no sigue la tendencia alcista prevista o se encuentra en un rango lateral.

El mercado utilizado en esta estrategia es conocido como mercado spot. En este tipo de mercado, se llevan a cabo transacciones directas de compra y venta de la criptomoneda en sí. Al realizar una orden de compra, se adquiere una cantidad específica de la criptomoneda a cambio de una cantidad equivalente de USDT (que representa un dólar). Por otro lado, al ejecutar una orden de venta, se vende la criptomoneda previamente adquirida y se obtienen unidades de USDT nuevamente.

Es importante destacar que en este enfoque de compra y venta, no existe el riesgo de liquidación, sino que el principal riesgo radica en la posibilidad de que el valor de la criptomoneda disminuya. Sin embargo, las criptomonedas adquiridas siempre permanecerán en posesión del inversor, lo que significa que no existe el riesgo de liquidación como ocurre en el mercado de futuros. Además, a diferencia del mercado de futuros, en el mercado spot no se utiliza apalancamiento, sino que se opera únicamente con el capital disponible.

En el método utilizado, se emplea el indicador de media móvil simple (SMA). Este indicador permite observar el comportamiento del precio en un determinado número de sesiones. Por ejemplo, si se utiliza una media móvil simple de dieciocho sesiones, se obtendrá el promedio del precio de los últimos dieciocho días. Al unir estos valores, se forma la línea de media móvil, cuya terminación varía de acuerdo a las fluctuaciones del precio. Este indicador proporciona una idea de la tendencia del mercado, permitiéndonos aprovecharla para obtener beneficios. Continuando con el ejemplo, si las SMAs toman el valor del activo al cierre, el cálculo de la media móvil simple se realiza de la siguiente manera:

SMA = Sumatoria de precios de cierre de las últimas 18 sesiones / 18

Además de la media móvil simple, existen otros tipos de medias que también son útiles en las operaciones, como la media móvil exponencial y la media móvil ponderada, las cuales consideran otros aspectos en su cálculo.

3.4 Gestión de Riesgo

En la inversión y operaciones especulativas, es esencial considerar la gestión de riesgos para preservar el capital a largo plazo. En la operativa del bot, se utiliza el

concepto de "unidades de riesgo", que consiste en dividir el capital total en cantidades utilizadas para comprar el activo en cuestión. Esto permite realizar múltiples operaciones y evitar concentrar el capital en unas pocas transacciones. El objetivo es preservar y hacer crecer el capital gradualmente, obteniendo ganancias consistentes a largo plazo. Cada vez que se utiliza una unidad de riesgo para comprar, se adquieren unidades del activo que permanecerán en posesión del inversor. Esta estrategia evita la liquidación de posiciones por parte del Exchange, protegiendo así el capital arriesgado en caso de que las operaciones no se desarrollen según lo analizado por el bot.

Es importante tener en cuenta que el precio de cualquier activo puede llegar a cero en cualquier momento. A pesar de esto, las unidades del activo adquiridas permanecerán en posesión del inversor, brindando la posibilidad de recuperar parte del capital u obtener beneficios en caso de que el precio se valorice nuevamente. Por lo tanto, el riesgo de pérdida es bajo, ya que se retienen las unidades adquiridas hasta que el precio actual supere el precio de entrada al mercado.

3.5 Funcionamiento

En cuanto al funcionamiento del bot de trading, se procede a desglosar las diferentes etapas del proceso:

1. Cuando se dispone de saldo para operar, el bot realiza una solicitud mediante la API de Binance para obtener los datos de un activo específico en el mercado de criptomonedas.
2. La API de Binance, a su vez, solicita los datos correspondientes al Exchange y recibe una respuesta detallada según la solicitud realizada.
3. El bot recibe los datos proporcionados por Binance y lleva a cabo los cálculos de los indicadores que utiliza para el análisis.
4. Posteriormente, el bot examina los resultados de los indicadores y, en función de dicho análisis, determina si es necesario realizar una compra, venta o no llevar a cabo ninguna acción.
5. En caso de que el bot determine que se debe comprar o vender, es decir, ejecutar una operación en el Exchange, envía la solicitud correspondiente a través de la API de Binance.
6. La API de Binance, a su vez, transmite la solicitud de compra o venta al Exchange.
7. Con esta petición, el Exchange realiza la acción solicitada por el bot.

Este proceso se repite de forma constante a intervalos regulares, configurados para mantener el estado del hardware utilizado. Es importante tener en cuenta que se debe ejecutar una instancia separada del bot para cada par de criptomonedas que se desee operar, lo que permite al bot llevar a cabo operaciones simultáneas en diferentes activos y aprovechar al máximo las oportunidades de trading.

4. Resultados Obtenidos

Las experiencias con el uso del bot de trading han sido satisfactorias para los casos de estudio de inversiones realizadas. A partir de la utilización de esta herramienta automatizada, se ha logrado realizar operaciones de compra y venta de criptomonedas de manera eficiente y rentable. El análisis técnico y las estrategias implementadas por el bot han generado resultados positivos y beneficiosos. Debe tenerse en cuenta, además, que las inversiones realizadas fueron muy limitadas en cuanto al volumen de dólares involucrados. El uso del bot ha permitido aprovechar oportunidades de compraventa en el mercado de criptomonedas, incluso en transacciones de menor escala. La capacidad del bot para ejecutar operaciones de forma continua y en tiempo real ha brindado ganancias consistentes a lo largo del tiempo. La efectividad del bot de trading ha demostrado ser una ventaja en la estrategia de inversión de los usuarios. Su desempeño confiable y los resultados positivos obtenidos han fortalecido la confianza en la toma de decisiones y han maximizado los beneficios en el mercado de criptomonedas.

Por último, se realizó un análisis FODA del proyecto. Este análisis estudia, descubre y entiende los aspectos positivos y negativos del proyecto. Para ello se analizan Fortalezas, Amenazas, Debilidades y Oportunidades.

Las fortalezas del proyecto:

- Requisitos tecnológicos bajos: un equipo con procesador Intel i5 8 GB de RAM y preferiblemente un disco SSD.
- Bajo costo: como corolario del anterior.
- Eliminación de emociones humanas: el bot permite eliminar las emociones humanas al operar.

Las oportunidades de mercado que pueden beneficiar el proyecto se centran en:

- Incorporación de nuevos pares de criptomonedas.
- Cambios en la API del Exchange: es posible que se agreguen nuevas funcionalidades que faciliten la operación o el intercambio de información con el Exchange.
- Nuevas estrategias: que adapten o mejoren el bot de trading y que surjan a partir del análisis de datos obtenido.

Las debilidades o puntos negativos del proyecto se centran en:

- Dependencia de internet.
- Contexto del mercado: el mercado cambia a lo largo del tiempo, lo que puede hacer que la estrategia requiera adaptación o cambio.
- Análisis fundamental: el bot se basa en el análisis técnico, dejando de lado el contexto y las noticias que afectan al mercado.

Por último, existen amenazas que podrían perjudicar el desarrollo del proyecto:

- Caída del servicio del Exchange.
- Mercados bajistas: la tendencia bajista del mercado puede prolongarse según el ciclo del mercado y el contexto mundial y económico.
- Fallo en el servidor virtual (VPS).
- Problemas logísticos: interrupción del servicio eléctrico y/o internet.

5. Conclusiones

A través de la investigación y el análisis realizado, se ha obtenido un mayor entendimiento y conocimiento sobre la relación entre los diferentes actores que interactúan en el sistema, así como su participación en los diversos casos de uso identificados.

Se constató que los actores involucrados en el sistema llevan a cabo acciones específicas de acuerdo con las indicaciones proporcionadas por el sistema, el cual se encarga de gestionar el flujo de información y datos intercambiados entre ellos.

En resumen, se logró obtener una comprensión precisa de las interacciones entre los actores y el sistema, y se ha evidenciado la importancia de seguir las indicaciones proporcionadas por este último para un correcto funcionamiento del sistema en su totalidad.

Estos logros demuestran que el estudio ha cumplido el objetivo de brindar conocimientos en relación a las interacciones de los actores y el sistema, proporcionando una base sólida para futuros avances y mejoras en el ámbito de las criptomonedas, blockchain y trading automatizado.

Referencias

1. Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System. Consultado en https://bitcoin.org/bitcoin.pdf
2. Antonopoulos, A. M. (2014). Mastering Bitcoin: Unlocking Digital Cryptocurrencies. O'Reilly Media.
3. Tapscott, D., & Tapscott, A. (2016). Blockchain Revolution: How the Technology Behind Bitcoin Is Changing Money, Business, and the World. Penguin.
4. Popper, N. (2016). Digital Gold: Bitcoin and the Inside Story of the Misfits and Million
5. Cheng, S., Meltzer, J., & Shi, X. (2019). Cryptocurrency Investment Strategies: A Comparative Analysis of Bitcoin, Ethereum, and Litecoin. Journal of Risk and Financial Management, 12(1), 25. doi:10.3390/jrfm12010025
6. Hayes, A. S., & Mihov, V. T. (2018). Cryptocurrencies: A Brief Review and Analysis of the Current Landscape. Economics: The Open-Access, Open-Assessment E-Journal, 12(2018-39), 1-22. doi:10.5018/economics-ejournal.ja.2018-39
7. Karame, G. O., Androulaki, E., Capkun, S., & Čapkun, S. (2015). Double-spending fast payments in Bitcoin. Proceedings of the 2015 IEEE Symposium on Security and Privacy, 204-219. doi:10.1109/SP.2015.18
8. Shrier, D. A., Forni, M. F., & Pentland, A. S. (2016). Trust, Distributed Ledgers, and Blockchain Technology. Trust: Data, A Decentralized Framework for Creating Secure
9. Applications, 51-80. doi:10.2139/ssrn.2839075
10. Hull, J. C. (2018). Options, Futures, and Other Derivatives (10th Edition). Pearson.
11. Elder, A. (2014). Trading for a Living: Psychology, Trading Tactics, Money Management. Wiley.
12. Pollini, Juan Pablo (2022). Bot de Trading. Trabajo final de Grado Universidad Católica de La Plata. Facultad de Ciencias Exactas e Ingeniería.
13. Downey, A. B. (2015). Think Python: How to think like a computer scientist. Green Tea Press.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Diseñar software accesible: pensar fuera de la caja

Claudia Ortiz[1], Cecilia Challiol[2], Walter Panessi[1]

[1] Departamento de Ciencias Básicas Universidad Nacional de Luján,
Luján, Buenos Aires, Argentina.
{cortiz, wpanessi}@unlu.edu.ar
[2] LIFIA, Facultad de Informática, UNLP and also CONICET,
La Plata, Buenos Aires, Argentina
ceciliac@lifia.info.unlp.edu.ar

Resumen. La accesibilidad es abordada generalmente como un aspecto de implementación, y no se involucran a las personas con discapacidad en las etapas tempranas de diseño. Por otro lado, el auge de Design Thinking ha llevado a replantearse si los métodos tradicionales de empatización (entrevistas y/o encuestas) llegan a captar las necesidades reales de las personas, en particular aquellas con discapacidad. En este trabajo se propone una guía para empatizar con personas con discapacidad dentro de experiencias de Design Thinking, para el diseño de software accesible desde su concepción. Usando esta guía, se diseña y explora la empatización con dos personas, una con desafíos motrices (que afectan su habla y su escritura) y otra con Síndrome de Down. Con este trabajo se busca reflexionar sobre cuáles son los mejores recursos para empatizar y aportar en la dirección de diseñar software accesible.

Palabras claves: Design Thinking, Software, Accesibilidad, Discapacidad, Diseño Centrado en las Personas.

1. Introducción

La accesibilidad generalmente se aborda en etapas avanzadas del desarrollo por considerarse un aspecto de implementación [1], lo cual genera una sobrecarga adicional de trabajo para lograr que el software sea accesible. Muchas veces el producto resultante no llega a ser usable o genera frustración a los usuarios con discapacidad [2], ya que el software no fue diseñado considerando sus necesidades reales [3]. Por este motivo, es crítico involucrar activamente a las personas con discapacidad en el diseño de software [4].

Design Thinking (DT) [5] es un abordaje de Diseño Centrado en las Personas, que utiliza los métodos de los diseñadores para crear soluciones innovadoras, que se ajusten a las necesidades de las personas. DT hace principal hincapié en lograr una primera etapa de empatía para identificar o descubrir qué necesitan realmente las personas. Más aun, la empatía podría favorecer a lograr un mundo más accesible [6]. Sin embargo, DT está poco explorado para diseñar software accesible [7], lo cual motiva nuestra línea de investigación.

La forma de empatizar puede ser mediante interacciones directas o indirectas [8]. Las interacciones directas implican recopilar información de "*primera mano*", por ejemplo, mediante cuestionarios y/o entrevistas a usuarios representativos [9]. Mientras que las interacciones indirectas pueden involucrar, por ejemplo, observaciones in-situ

a usuarios representativos o entrevistas a terceros [10]. Hay que tener en cuenta que las interacciones indirectas pueden estar sesgadas y no reflejar las vivencias reales de las personas con las cuales se quiere empatizar [8].

Para la etapa de empatización de DT se pueden utilizar distintos recursos, los cuales pueden implicar interacciones directas o indirectas. Sin embargo, es recomendable abordar esta etapa con interacciones directas, y las indirectas usarlas como complementarias [7]. Por otro lado, los recursos tradicionales usados en interacciones directas de DT, como son las entrevistas o cuestionarios, pueden no llegar a generar una empatía real por las personas con las cuáles se está buscando empatizar [11]; es decir, no se identifican sus necesidades reales. Esto es parte de la motivación del presente trabajo.

En [7] hemos presentado un framework conceptual de DT para el diseño de software accesible desde su concepción. Además, en [7] se ha explorado la empatización con personas daltónicas para comenzar a comprender cuáles son sus necesidades reales en relación con el software. El presente trabajo se focaliza en enriquecer el framework conceptual propuesto en [7], y además contribuir con la exploración de empatización con otras discapacidades.

En este trabajo se presenta una guía para la empatización con personas con discapacidad con foco en interacciones directas; esta guía se enmarca en el framework propuesto en [7]. Usando la guía, se diseña y explora la etapa de empatización con una persona con desafíos motrices, que afectan su habla y su escritura; y con una persona con Síndrome de Down.

Además, se espera generar un espacio de discusión que contribuya a visibilizar la problemática planteada, buscando reflexionar sobre cómo detectar las necesidades reales de las personas con discapacidad, para luego poder diseñar adaptaciones en el software desde etapas tempranas, y lograr que el mismo sea accesible desde su concepción. Para esto, creemos que es necesario "*pensar fuera de la caja*" sobre los métodos usados para diseñar software accesible.

Este trabajo tiene la siguiente estructura. En la Sección 2 se describen algunos trabajos relacionados. En la Sección 3 se presenta la guía propuesta. En la Sección 4 se enuncian dos experiencias de empatización. En la Sección 5 se plantea un espacio de discusión. Las conclusiones y trabajos futuros se mencionan en la Sección 6.

2. Trabajos Relacionados

En los últimos años se ha comenzado a utilizar DT para diseñar soluciones de software. Para el presente trabajo es de interés conocer cómo distintas investigaciones abordan en particular la etapa de empatización de DT con personas con discapacidad. A continuación, se describen dos trabajos en relación con esta temática.

En [10] se usa DT para el diseño y desarrollo de una interfaz de usuario tangible para el proceso de alfabetización de estudiantes con Síndrome de Down. Para la etapa de empatización de DT, entrevistaron a quince maestras de Educación Especial, lo cual permitió identificar dos métodos de lectoescritura utilizados con estos alumnos. Luego, se profundizó en cada método, entrevistando a sus creadoras, y además se realizó una revisión de literatura para complementar la información. Por último, los autores

llevaron a cabo una observación in-situ de las maestras trabajando con los estudiantes con el objetivo de entender la aplicación de los métodos. Es decir, para la empatización se utilizan todos recursos que implican interacciones indirectas.

Por otro lado, en [9] se presenta una experiencia de DT con una empresa de transporte, con el objetivo de mejorar el servicio para personas con desafíos en la movilidad. Para la etapa de empatizar se realizaron las actividades que se describen a continuación. El autor investigó qué les pasa a los usuarios de transporte (usando *Netnografía*[1]), cómo actuaban las empresas (utilizando *Benchmarking*[2]) y cómo es la vivencia de un usuario del transporte con el objetivo de mejorarla (mediante el recurso *Safari*[3]). Se realizaron cuestionarios y entrevistas con los usuarios del transporte. La información obtenida se organizó usando distintos recursos[4] (User Persona, User Journey y Mapa de Empatía). En resumen, en [9] solo los cuestionarios y entrevistas son recursos que implican interacciones directas; sin embargo, estos recursos podrían no haber logrado una empatía real [11].

A continuación, se describen dos trabajos que exploran cómo personas con discapacidad (con Síndrome de Down y con desafíos motrices) usan gestos en aplicaciones móviles. Estos trabajos no usan DT, pero son interesantes en cuanto a la exploración que se hace en relación con el software.

En [12] se presenta una investigación realizada junto con quince jóvenes con Síndrome de Down, respecto a la utilización de los gestos más básicos en las aplicaciones móviles. La investigación divide a los jóvenes en tres grupos, todos tenían nulo o poco uso de dispositivos móviles. Al primer grupo le dieron seis aplicaciones que sirven de entrenamiento de seis gestos distintos. El segundo grupo recibió solo una explicación sobre los gestos. Mientras que el tercer grupo no tuvo ningún tipo de información en relación con los gestos. Luego, los tres grupos utilizaron la aplicación Google Earth (seleccionada por tener una interfaz poco intuitiva en cuanto al uso de gestos). En [12] se menciona que las aplicaciones de entrenamiento han facilitado el uso de Google Earth. Los autores identifican que los gestos de *rotar*, *mantener presionado* y *arrastrar* son los más difíciles de realizar por estos jóvenes, aunque no describen cómo llegan a esta conclusión.

Por otro lado, en [13] se estudian las dificultades que generan los gestos en aplicaciones móviles con diez personas con distintos desafíos de movilidad reducida en los miembros superiores. Para esto, los autores relevan los gestos más usados en

[1] La *Netnografía* se usó para investigar en redes sociales, páginas web y foros interacciones escritas de personas que usan o han usado el transporte público y han expresado en ellas sus sentimientos, emociones y experiencias.

[2] *Benchmarking* se realizó con un ejercicio denominado "busca las 7 diferencias" con el que investigaron autobuses de otras comunidades y países.

[3] El recurso *Safari* consiste en una simulación in-situ que permite vivenciar las experiencias de las personas con las cuales se está buscando empatizar.

[4] Se especificó el recurso "*User Persona*" con los datos recopilados para representar una persona ficticia que utiliza el autobús, y que tiene dificultades de movilidad. Luego, con el recurso "*User Journey*" se determinaron los puntos fuertes/débiles para el perfil definido en "*User Persona*". Finalmente, en base a toda la información recolectada, con el recurso "*Mapa de empatía*" se reflejaron los aspectos emocionales en relación con el sentir y pensar de los usuarios del autobús.

aplicaciones existentes. En base a esto, desarrollaron un prototipo programado para iPhone, con el objetivo de saber qué gestos no pueden hacer las personas con distintos desafíos de movilidad. Luego de usar el prototipo, las personas con movilidad reducida completaron un cuestionario con una serie de preguntas sobre la usabilidad del mismo. En [13] se identifica para las diez personas que participaron del uso del prototipo, en base a las respuestas del cuestionario, que los gestos más difíciles de realizar fueron *rotar* y *agrandar*.

Tanto en [12] como en [13] se evidencia que las exploraciones se focalizan puramente en la usabilidad de los gestos, y no se empatiza realmente con las personas.

3. Guía para empatizar con personas con discapacidad

Como se mencionó anteriormente, en [7] hemos presentado un framework conceptual de DT para el diseño de software accesible desde su concepción. Este framework propone tres etapas para abordar DT como son: empatizar, idear y prototipar/testear. Cabe mencionar que en [7] no se detalla ninguna guía para abordar estas etapas.

Para la etapa de empatizar del framework conceptual de DT propuesto en [7], se propone a continuación, una guía de ítems a considerar para empatizar con personas con discapacidad con foco en interacciones directas:

- *Elegir una persona empática para conducir la etapa de empatización*. Es fundamental considerar el nivel de empatía que tiene la persona que va a llevar a cabo la etapa de empatización, para lograr una comunicación más amena y favorecer una escucha activa de las necesidades de las personas con discapacidad. Existen varios test para medir el nivel de empatía [7].
- *Generar un vínculo de confianza con la persona que se desea empatizar*. La existencia o no de un vínculo previo, determina cómo proseguir. Si no se conoce a la persona, hay que generar ese vínculo de confianza, para luego poder llevar a cabo cualquier actividad de empatización en relación con el software. Para esto se puede utilizar, por ejemplo, el recurso denominado "*Entrevista para Empatizar*" [14]. Ya sea que exista un vínculo previo o que se haya generado, hay que contarle a la persona, con la que se está empatizando, en qué consiste la experiencia que se va a llevar a cabo y por qué es de suma utilidad su aporte; junto con un documento de "*Consentimiento Informado*" (que brinda todos los detalles de la experiencia).
- *Determinar la comunicación preponderante de la persona con la que se desea empatizar*. Conocer cómo las personas se comunican se vuelve crítico para tener una mejor llegada a éstas, y comprender sus necesidades. Las personas pueden ser visuales, auditivas o kinestésicas (aprenden con lo que hacen y tocan), o una combinación de éstas [15]; esto impacta en la forma en que se expresan y/o incorporan información. Se puede determinar la comunicación preponderante, por ejemplo, mediante el recurso de "*Observación Directa*", y en base al resultado, adecuar dinámicamente la selección de los recursos para empatizar con cada persona de la forma más adecuada.
- *Conocer respecto a la discapacidad de la persona con la que se desea empatizar.* Es de vital importancia conocer respecto a cómo impacta la discapacidad de la persona

en su proceder diario. Dependiendo de la cercanía del entrevistador con la discapacidad de la persona, es el nivel de profundidad con la que tiene que explorar la misma. En este sentido se puede conocer de la discapacidad a través de: preguntas hechas a terceros (expertos, familiares), literatura, observación, etc.

- *Definir qué indagar respecto al tipo de software que se está queriendo diseñar.* Cada tipo de software puede presentar distintas características y dinámicas, definir cuál se desea diseñar es fundamental para orientar cómo llevar a cabo la etapa de empatización en relación con el software y qué recursos utilizar para la misma.
- *Determinar qué recursos utilizar para empatizar en relación con el software que se está queriendo diseñar.* Se pueden generar recursos que permitan empatizar en relación con el software, por ejemplo, un bosquejo (mock-up) de algún aspecto relevante del software; este estilo de recursos es fácil de diseñar/armar y no demanda mucho tiempo. Otra opción es usar un "*Prototipo para Empatizar*"[5] [14].

4. Experiencias de Empatización

En esta sección se presentan dos experiencias de puesta en práctica de la guía propuesta en la Sección 3, con dos personas con discapacidad[6]. Una de estas personas tiene desafíos motrices, que afectan la claridad del hablar y de su escritura, de ahora en más se mencionará como T. La segunda persona tiene Síndrome de Down, de ahora en adelante J. Ambas personas usan habitualmente dispositivos móviles, tienen adquirida la lectoescritura, participan en grupos de teatro y baile respectivamente, y en el caso de J además tiene un trabajo. A continuación, se van a ir abordando cada uno de los ítems propuestos en la guía para ambas experiencias, las cuales acontecieron en paralelo y de forma personalizada con cada persona.

Para "*Elegir una persona empática para conducir la etapa de empatización*", fue necesario considerar cuál era la persona del equipo de trabajo más empática para conducir la experiencia, para esto se utilizó el test de IRI[7]. Dos de los integrantes del equipo dieron como resultado en la dimensión de preocupación empática, un valor superior a la media, de los cuales se eligió el que tenía más vinculo de cercanía con las personas con las que se quería empatizar. De ahora en adelante esta persona será denominada entrevistadora. En la Tabla 1, se describe cómo fueron abordados tres de los ítems de la guía.

[5] El "*Prototipo para Empatizar*" tiene funcionalidad (es decir se puede interactuar), puede ser tanto un software creado para tal fin o uno existente. El concepto de "*Prototipo para Empatizar*" es diferente al concepto de "*Prototipo de Software*", ya que se crea o usa solo con el fin de empatizar, y no se convertirá en el software final.

[6] Para preservar la identidad de las personas involucradas, serán identificadas con una letra.

[7] El Test de IRI (Índice de Reactividad Interpersonal) permite medir empatía en adultos. Versión on-line para autoevaluación: www.idiena.com/test/IRI (último acceso: 12/05/2023).

Tabla 1. Puesta en práctica de tres ítems de la guía propuesta para empatizar.

Ítem de la Guía	T (Desafío Motriz y en el habla)	J (Síndrome de Down)
Generar un vínculo de confianza con la persona que se desea empatizar	La entrevistadora conocía a la persona previamente. Sin embargo, hacía tiempo que no tenían contacto, hubo que retomar el vínculo.	La entrevistadora no conocía a J. Se decidió realizar una "*Entrevista para Empatizar*" [14] presencial, con la asistencia de una tercera persona que fue el nexo entre la entrevistadora y J; esta persona (de ahora en más R) cumple un rol similar a un tutor de J.
Determinar la comunicación preponderante de la persona con la que se desea empatizar	La entrevistadora lo fue refinando durante la re-vinculación con T, la cual se dio primero contactándose por Whatapps y luego mediante una videollamada por Meet (configurado con la opción de subtítulos, por la dificultad del habla que tiene T).	La entrevistadora lo fue refinando mediante la "*Observación Directa*" durante la "*Entrevista para Empatizar*".
Conocer respecto a la discapacidad de la persona con la que se desea empatizar	Como la entrevistadora conocía a T, sabia como su discapacidad le impactaba puntualmente en su vida diaria.	Se conoció sobre el Síndrome de Down a través de consultas a terceros: a una experta (de ahora en más E) para saber respecto a las eneralidades, y con R para saber en particular sobre J.

En cuanto a *"Definir qué indagar respecto al tipo de software que se está queriendo diseñar",* se decidió indagar sobre la utilización de gestos en aplicaciones móviles tanto para T como para J, motivados por el uso frecuente de dispositivos móviles y por la escasa empatización directa que se ha encontrado en la literatura. En particular, se decidió explorar sobre los gestos: *rotar*, *agrandar* y *arrastrar* (que implica *mantener presionado*); considerando que pueden generar desafíos, como se mencionó en la Sección 2, para las dos discapacidades exploradas en este trabajo.

Para el ítem de la guía "*Determinar qué recursos utilizar para empatizar en relación con el software que se está queriendo diseñar*", se eligieron los recursos que se listan en la Tabla 2.

Tabla 2. Recursos utilizados para empatizar en relación con el software.

Recurso Usado	Descripción	T (Desafío Motriz y en el habla)	J (Síndrome de Down)
Prototipo para Empatizar	Usar una aplicación conocida.	Word Wonders[8]	Candy Crush[9]
Prototipo para Empatizar	Edición de una imagen de Whatsapp.	Contextualizada al teatro (un tema de interés de T)	Contextualizada a caballos (un tema de interés de J)

En dos encuentros presenciales, uno con cada persona, primero se les propuso utilizar de forma libre una aplicación conocida como "*Prototipo para Empatizar*", la cual permitió a la entrevistadora realizar una "*Observación Directa*" de cómo utilizan los dispositivos móviles habitualmente. Las aplicaciones *Word Wonders* y *Candy Crush* fueron elegidas considerando la información obtenida en las interacciones previas.

Luego de unos minutos, la entrevistadora le realizó a cada uno algunas preguntas relacionadas con la complejidad para usar cada aplicación. T a diferencia de J contestó las preguntas por Whatsapp, aunque él y la entrevistadora se encontraban en el mismo lugar físico; esto se decide así para saltar la barrera de la dificultad en la comunicación oral que tiene T. La entrevistadora observó que ninguno tuvo problemas, es más lo usaron fluidamente; sin embargo, T mencionó "*a veces cuando levanto el dedo me es difícil*", por otro lado, T hizo foco en que lo más difícil era a veces adivinar la palabra (ya que usó *Word Wonders*).

Posteriormente, se les propone editar una imagen en blanco usando Whatsapp, en este caso se aprovecha una aplicación que utilizan frecuentemente para probar el uso de gestos de una forma guiada. Esta actividad de edición es la misma para ambas personas, lo único que varía es la temática usada para cada una ya que se contextualizó acorde a sus intereses, como se puedo apreciar en la Tabla 2 (los cuales fueron identificados por la entrevistadora en las interacciones previas).

Para la edición, se les fue indicando paso a paso lo que debían ir haciendo y se les mostraba una imagen de cómo les debía quedar, con poca indicación del cómo ir haciéndolo. Esta secuencia estuvo pensada en función de los gestos que se deseaban observar, por ejemplo, se les pidió agregar emojis y textos, ubicarlos en determinados lugares con cierta inclinación y tamaño.

En la Figura 1 se puede observar una imagen general de la situación de la experiencia de empatización, en relación con los gestos, realizada con T. Además, se pueden apreciar las pantallas finales, tanto de la entrevistadora como el resultado final obtenido por T luego de seguir los pasos de la edición; la cual es similar a lo propuesto por la entrevistadora, en distribución y tamaño de los emojis y texto.

[8] Word Wonders: https://play.google.com/store/apps/details?id=com.fugo.wow (último acceso: 3/7/2023)

[9] Candy Crush: https://play.google.com/store/apps/details?id=com.king.candycrushsaga (último acceso: 3/7/2023)

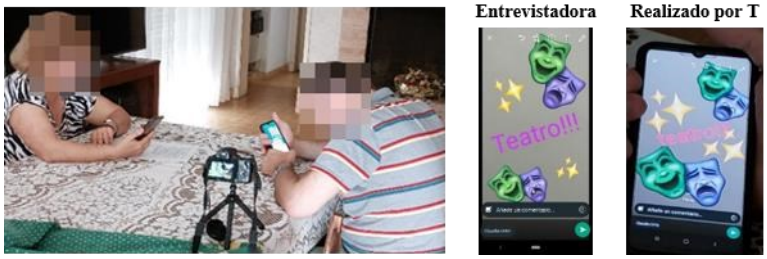


Fig. 1. Experiencia realizada con T, para empatizar en relación con los gestos.

Después que finalizó la edición, se le preguntó a T respecto a qué complejidad tuvo para rotar, agrandar y arrastrar los emojis. T contestó por Whatsapp en los tres casos "*no tuve problema*", esto se condice con lo que observó la entrevistadora.

En la Figura 2 se muestra la situación de la experiencia realizada con J; cabe destacar que J se dispersó constantemente durante la experiencia. Además, en la figura se observa la pantalla final mostrada por la entrevistadora y el resultado obtenido por J al finalizar la edición. Se puede observar una diferencia significativa en tamaños e inclinación tanto de los emojis como del texto.



Fig. 2. Experiencia realizada con J, para empatizar en relación con los gestos.

Después que finalizó la edición, se le preguntó a J sobre qué complejidad tuvo para cada gesto (*rotar*, *agrandar* y *arrastrar*). Ante el planteo de la primera pregunta, parecía que J no entendía a qué iba orientada, entonces la entrevistadora la reformuló brindándole opciones: *"¿Qué tan complejo te fue rotar los emojis? te fue fácil, difícil o muy difícil*"; entonces J contestó "*Muy difícil, ahí de difícil, más o menos*". La entrevistadora formulo las siguientes preguntas brindando las mismas opciones, y las respuestas de J fueron del mismo estilo, mencionando distintos grados de complejidad en una misma respuesta. Esto llevó a la entrevistadora a no saber cómo interpretar estas respuestas.

Estas dos experiencias, planificadas usando la guía propuesta, son un primer paso para explorar la empatización con personas con discapacidad con foco en las interacciones directas. En la siguiente sección se discuten distintos aspectos de interés en relación con los emergentes que acontecieron en estas experiencias.

5. Discusión

En esta sección se busca reflexionar sobre cuáles son los mejores recursos para empatizar con foco en interacciones directas, y aportar en la dirección de diseñar software accesible desde su concepción.

Elegir una persona empática para conducir la experiencia no es una consideración habitual de DT [5]; sin embargo, hemos identificado que al interactuar con personas con discapacidad esta habilidad se vuelve fundamental para tener una escucha activa de sus necesidades reales.

Por otro lado, la generación del vínculo de confianza con la persona que se desea empatizar es fundamental para sentar las bases de toda la interacción que se dará a posterior. Como se ha mencionado anteriormente, los recursos como entrevistas o encuestas no llegan a generar una empatía real [11]. Es por lo que recomendamos usar la "*Entrevista para Empatizar*" [14], que es un recurso catalogado dentro de los posibles a utilizar dentro de una experiencia de DT, aunque en la mayoría de los casos no se aplica. Este tipo de entrevista está diseñada para que las personas, con las que se desea empatizar, cuenten lo que es realmente importante para ellos; esto en muchos casos puede escapar a aspectos de software, sin embargo, esta información es muy valiosa ya que puede impactar en el uso del mismo.

Determinar la comunicación preponderante de la persona con la que se desea empatizar es otro ítem fundamental, en nuestras experiencias se realizó mediante la "*Observación Directa*". En el caso de J, la entrevistadora detectó que se dispersaba constantemente, lo cual requirió que la misma tuviera una postura flexible en todo momento. Por otro lado, la situación de T era distinta porque la entrevistadora conocía cómo su discapacidad impactaba en su habla y escritura, se comenzó la comunicación por Whatsapp desarrollándose sin problemas, luego se probó el uso de Meet con subtítulos activados. Sin embargo, Meet no fue de ayuda porque la herramienta no lograba descifrar lo que decía T, esto llevó a tomar la decisión de recolectar las respuestas a las preguntas con relación al uso de gestos mediante Whatsapp; así T pudo expresarse sin ninguna barrera.

Con relación a conocer respecto a la discapacidad de la persona con la que se desea empatizar, se reflexionó que se hubiera necesitado indagar más inicialmente sobre el Síndrome de Down; esto se evidenció con las respuestas ambiguas que se obtuvieron de J (en la Sección 4). Posteriormente a la experiencia con J, la entrevistadora consultó nuevamente a los terceros mencionados en la Tabla 1; identificando que el concepto de complejidad puede ser muy abstracto aun habiéndole brindado opciones en la pregunta. Además, en esta nueva indagación se detectó la importancia, para las personas con Síndrome de Down, de la observación y repetición para incorporar o usar algo nuevo; lo cual se condice con los beneficios del entrenamiento en gestos mencionado en [12]. Un aspecto para reflexionar es que muchas veces se busca el primer impacto de la persona con el software; sin embargo, con las personas con Síndrome de Down debería considerarse el entrenamiento como un aspecto positivo.

Es importante destacar que a T no le fue difícil realizar los gestos que se planteaban en [13], es decir que cada persona tiene distintas necesidades. Esto visibiliza que se requieren más exploraciones de empatización para poder recolectar qué adaptaciones son necesarias considerar al diseñar software accesible.

6. Conclusiones y Trabajos Futuros

Para diseñar software accesible desde su concepción se vuelve fundamental incorporar recursos adecuados para empatizar con las personas con discapacidad e identificar sus necesidades reales, como, por ejemplo, las "*Entrevistas para empatizar*" [14] o los "*Prototipos para Empatizar*" [14]. En este sentido, creemos que nuestra guía para empatizar con foco en interacciones directas con personas con discapacidad podría ser de mucha ayuda. Además, se espera que este trabajo sirva para reflexionar sobre la temática y "*pensar fuera de la caja*" cuando se aborda este tipo de diseño. Por ejemplo, en muchos casos no se requiere de aplicaciones especializadas para establecer una comunicación, sino que con una buena indagación se puede detectar y aprovechar lo que ya usa habitualmente la persona, como fue el caso de T con Whatsapp. Otro cambio de mentalidad se da con relación a la concepción habitual de las pruebas de software sin sesgo inicial, ya que el entrenamiento puede ser muy positivo con personas con Síndrome de Down [12].

Este trabajo puede aportar a reflexionar sobre cómo realizar testeo de software con personas con discapacidad, ya que en muchos casos las aplicaciones terminan siendo probadas por terceros, que deciden sobre la utilidad de las mismas [8].

En cuanto a los trabajos futuros, se continuará explorando la empatización con más personas, para enriquecer la guía propuesta y cubrir las tres etapas del framework conceptual propuesto en [7] para el diseño de software accesible desde su concepción.

Referencias

1. Oliveira, R., Silva, L., Leite, J.C.S.P., Moreira, A.: Eliciting accessibility requirements an approach based on the NFR framework. In: Proceedings of the 31st Annual ACM Symposium on Applied Computing, pp. 1276-1281. ACM, New York (2016)
2. Kahraman, E.: Inclusive Design Thinking: Exploring the obstacles and opportunities for individuals and companies to incorporate inclusive design. KTH Royal Institute of Technology EECS, Sweden (2020)
3. Molina López, J., Medina, N.: Un enfoque para el diseño inclusivo de videojuegos centrado en jugadores daltónicos. Revista de la AIPO 2(1), 25-37 (2021)
4. Spiel, K., Gerling, K., Bennett, C., Brulé, E., Williams, R., Rode, J, Mankoff, J.: Nothing about us without us: Investigating the role of critical disability studies in HCI. In: Proceedings of Extended Abstracts of the 2020 CHI Conference on Human Factors in
5. Computing Systems, pp. 1-8. ACM, New York (2020)
6. Brown, T.: Design thinking. Harvard Business Review 86, 84-95 (2008)
7. Da Silva, J., Goncalves Ferreira, M.: For an inclusive design. Brazilian Journal of Development 6(7), 44878-44888 (2020)
8. Ortiz, C., Challiol, C., Panessi, W.: Design Thinking para el diseño de software accesible desde su concepción. In: VI IEEE Biennial Congress of Argentina (ARGENCON), pp. 1-8. IEEE Press, Argentina, San Juan (2022)
9. Bennett, C., Rosner, D.: The promise of empathy: design, disability, and knowing the other. In: Proceedings of the 2019 Conference on Human Factors in Computing Systems, Paper 298, pp. 1-13. ACM, New York (2019)
10. Barreiro, A.: Design Thinking to improve urban transport in Madrid. UX Collective (2018)

11. Ramos Galarza, C., Jadán Guerrero, J.: Innovación tecnológica para mejorar el proceso de lectura inicial en estudiantes con Síndrome de Down. Universidad Tecnológica Indoamérica, Ecuador, Ambato (2018)
12. Rylander Eklund Roth, K., Globocnik, D., Rau, C., Neyer, A.: Living up to the expectations: The effect of design thinking on project success. Creativity and Innovation Management 29(4), 667-684 (2020)
13. Méndez-Muñoz, E. S., Rentería-Ramírez, J. E., Flores-Ibarra, F., Mendoza-González, A., Álvarez, F., Camacho-González, M. F. Y., Aguilar-Navarrete, P.: Estudio del uso de gestos en interfaces móviles por personas con Síndrome de Down. In: V Congreso Nacional de Tecnologías en la Educación, Benemérita Universidad Autónoma de Puebla, México (2015)
14. López-Quesada, J.: Estudio y mejora de la usabilidad de aplicaciones móviles para pacientes con disminución de movilidad. Trabajo fin de grado, Escuela Politécnica Superior, Universidad de Jaen (2019)
15. Doorley, S., Holcomb, S., Klebahn, P., Segovia, K., Utley, J.: Design thinking bootleg. d. school, Stanford University (2018)
16. Lakin, D.: Vender con PNL una ventaja oculta. Editorial Sirio, España (2007)

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

A Prototypical Tool for Analyzing Functional Dependencies Induced from Spreadsheets

Sergio Alejandro Gómez[1,2] and Pablo Rubén Fillottrani[1,2]

[1] Laboratorio de I+D en Ingeniería de Software y Sistemas de Información (LISSI)
Departamento de Ciencias e Ingeniería en Computación
Universidad Nacional del Sur
San Andrés 800, (8000) Bahía Blanca, Argentina
{sag,prf}@cs.uns.edu.ar
[2] Comisión de Investigaciones Científicas de la Provincia de Buenos Aires

Abstract. We present an extension to the GF framework for OntologyBased Data Access with the aim of determining the functional dependencies that hold in a spreadsheet. Spreadsheets are restricted to a single table expressed as a CSV text file. An initial set of tentative functional dependencies is computed using the TANE datamining algorithm. This set is then presented to the user who is used as an oracle to revise it. Given a functional dependency, the user can see the tuples from the spreadsheet justifying it. The user can revise the validity of the functional dependency with the help of our system, which will generate tuples not present in the dataset by using values already present in the table. The user can then add some of the new records to the table when he considers their feasibility and rerun the miner to see if the functional dependency still holds. We present a running example along with a downloadable JAVA-based application with source code of the miner in the C programming language and the files used in our experiments to help with the reproducibility of our results.

Keywords. Spreadsheets, TANE, Functional dependencies, Databases.

1. Introduction

Spreadsheets are an essential tool for organizations as they provide a simple and flexible way to store, analyze, and visualize data. They allow for efficient data management, calculations and data analysis, enabling informed decision-making. They also support collaboration, allowing multiple users to work together on the same data set. Data in spreadsheet tables can sometimes face challenges in terms of organization due to several reasons. Firstly, spreadsheet tables lack the structure and strict constraints of a database, making it easier for inconsistencies and errors to occur. Without predefined data types and constraints, it becomes more challenging to ensure data integrity and enforce data organization rules.

Thus, while spreadsheets offer convenience and flexibility, they are not inherently optimized for data organization. To overcome these challenges, organizations often rely on more robust data management systems, such as databases, that provide structured schemas, data validation, and stronger organization capabilities. Normalized databases play a crucial role in data management by reducing redundancy and improving data integrity. The discovery of functional dependencies is an essential step in the normalization process. Functional dependencies are relationships between attributes in

a database. They identify the dependencies between the values of one set of attributes on another set of attributes. By discovering and analyzing these dependencies, we can identify the key determinants in a dataset and eliminate data redundancy. Normalization helps in achieving a more efficient and organized database structure. It eliminates data anomalies such as update, insertion, and deletion anomalies that can occur due to redundant or inconsistent data. Thus, the discovery of functional dependencies plays a crucial role in achieving these benefits and establishing a well-structured and optimized database design.

Algorithms for the discovery of functional dependencies, such as TANE (Topological Attribute Noise Elimination) [1], are essential tools in data management and database design. They play a crucial role in identifying and understanding the relationships between attributes in a dataset, enabling data cleaning and normalization processes and assisting in the identification of candidate keys, which are essential for designing well-structured relational databases.

In this paper, our goal is to tackle the problem of finding functional dependencies in a spreadsheet table. In brief, our semi-automatic proposal consists of: first, the spreadsheet table is represented as single datasource D comprised of a CSV plain text file with fields separated by either commas or semicolons; second, we run the TANE datamining algorithm over D to obtain a set S of candidate functional dependencies supported by the data; third, the set S is presented to the user who will act as an oracle in determining the viability of each candidate functional dependency in S. For this, the user can select a particular functional dependency f and the system shows the projection P of the current tuples (restricted to the fields referenced in f) in D supporting the functional dependency and also generates a new set N of potential tuples that are not currently present in D. The user can select a subset S_N of N containing some of these potential tuples add them to the tuples already present in D, building in fact a new set $P \cup S_N$. The TANE algorithm is then executed against $P \cup S_N$ for determining if f is still valid and what new functional dependencies N_f are discovered. The user has now the chance of determining if he wants to delete f from S and also if he wants to add some of the elements of N_f to the set S of candidate functional dependencies. Thus, the main contribution of this work is (i) proposing a method for determining a set of functional dependencies from a single spreadsheet table; (ii) provinding a functional prototypical implementation of the proposed approach, comprised of JAVA stand-alone program integrated into our GF framework for ontology based- data access; (iii) a set of examples to show how the approach works, and (iv) a modification of a third-party implementation of the TANE algorithm for being used independently from our system. We include source code and a functional executable file published online along with the data files to reproduce the results presented here. A running example is provided to illustrate our approach.

The rest of the paper is structured as follows. In Sect. 2, we briefly revisit the notion of CSV file. In Sect. 3, we discuss how to find functional dependencies in CSV files and the modifications we made to a third-party data miner published online. In Sect. 4, we present the module added to GF to revise functional dependencies in the CSV file. In Sect. 5, we review related work. In Sect. 6, we conclude and foresee future work.

2. Spreadsheets as CSV Files

Excel is a popular spreadsheet software developed by Microsoft. They consist of multiple worksheets, each containing a grid of cells organized into rows and columns. These cells can hold various types of data, such as numbers, text, formulas, and functions. Excel spreadsheets offer a wide range of features and functionalities for data manipulation, analysis, and visualization, including formatting options, sorting and filtering capabilities, and chart creation. In Fig. (1.a), we show a simple Excel table for representing information of owners of cars along with information of the cars themselves. The information of the owners is comprised of the columns *IDPerson* and *Name*. Cars are represented by the columns *IDCar*, *CarBrand* and *CarPerception*. Notice that the column *CarPerception* represents the perception of the particular brand by a particular user.

	A	B	C	D	E
1	IDPerson	Name	IDCar	CarBrand	CarPerception
2	1	John	1	Aston Martin	high
3	2	Mary	2	Fiat	low
4	3	Paul	3	Audi	medium
5	1	John	4	BMW	medium
6	1	John	5	Chevrolet	low

(a)

```
IDPerson;Name;IDCar;CarBrand;CarPerception
1;John;1;Aston Martin;high
2;Mary;2;Fiat;low
3;Paul;3;Audi;medium
1;John;4;BMW;medium
1;John;5;Chevrolet;low
```

(b)

Fig.1. (a) Spreadsheet for owners of cars and (b) CSV code of the spreadsheet

CSV files are a simpler and more universally compatible format for tabular data. CSV files are plain text files that store tabular data as a series of values, with each value separated by a comma. Each line in the file represents a row of data, and the comma acts as a delimiter to separate individual values within each row. CSV files do not support formatting, formulas, or multiple worksheets like Excel spreadsheets. However, they are lightweight, easy to read and write, and can be opened by various software applications, making them widely used for data exchange between different programs and systems. For instance, in Fig. (1.b), we present the CSV version of the Excel table of Fig. (1.a). Notice that in this particular case, the field separator is the semicolon character.

In previous publications ([2] and references there in), we have been reporting about the development of a framework for Ontology-Based Data Access called GF. In this work, we extend such framework with the aim of solving the problem of finding functional dependencies hidden in CSV tabular data. The proposal presented in this work has been integrated with such application. A closedsource stand-alone JAVA application is available online[1] to reproduce the results presented here. Also an open source extension of the TANE miner described in the next section is available for downloading and use.

3. Finding Functional Dependencies in Spreadsheets

[1] See http://cs.uns.edu.ar/~sag/cacic2023

In the context of relational databases, a functional dependency is a relationship between two sets of attributes within a database table, that describes the dependence of one set of attributes (known as the dependent attributes) on another set of attributes (known as the determinant attributes). That is, given the values of the determinant attributes, the values of the dependent attributes can be determined. Functional dependencies play a crucial role in database design and normalization, helping to ensure data integrity and minimize data redundancy.

TANE (Topological Attribute Noise Elimination) [1] is an algorithm used for mining functional dependencies from a given relational database table. It is based on the concept of the topological sort and utilizes pruning techniques to efficiently discover all the non-redundant functional dependencies within the table. A brief explanation of the TANE algorithm is as follows:

1. *Input:* The algorithm takes a relational database table as input, consisting of a set of attributes and their corresponding values.
2. *Candidate Generation:* Initially, TANE starts with a set of candidate functional dependencies that include individual attributes and pairs of attributes. For example, if the table has attributes A, B, and C, the initial set candidates would be $\{A \rightarrow B, A \rightarrow C, B \rightarrow A, B \rightarrow C, C \rightarrow A, C \rightarrow B\}$.
3. *Pruning:* The algorithm employs pruning techniques to eliminate redundant candidates. It checks if each candidate can be further extended by adding more attributes without violating the closure property. If a candidate is found to be redundant, it is removed from the set of candidates.
4. *Topological Sorting:* TANE performs a topological sorting of the attributes based on their dependencies to ensure that dependencies are discovered in a particular order. It determines the dependencies between attributes by computing the closures of attribute sets. The closure of an attribute set is the set of all attributes that can be determined based on the given set of attributes.
5. *Dependency Discovery:* TANE iterates through the topologically sorted attribute order and discovers functional dependencies by checking if each candidate is satisfied by the current set of attributes. If a candidate is satisfied, it is considered a valid functional dependency and added to the result set.
6. *Closure Pruning:* After discovering each dependency, TANE applies closure pruning to eliminate any remaining redundant candidates that are no longer necessary based on the dependencies found so far.
7. *Repeat Step*: Steps 4 to 6 are repeated until no more dependencies can be discovered.
8. *Output:* The final result set contains all the non-redundant functional dependencies that have been mined from the input table.

Our approach to finding the functional dependencies holding in a spreadsheet starts by running a TANE miner on the CSV file contents. For doing this, we adapted an already existing TANE implementation.[2] That implementation is a console application based on the C programming language. It takes as input a CSV file and as output it

[2] See https://github.com/getterk96/Database-Functional-Dependency-Digging-Algorithms.

prints to the screen the functional dependencies that it has found. The dependencies found are listed as field indexes.

From that starting point, we extended that miner for having a more friendly interface that allows to use it as parameterized command-line application where the input file, output file, temporary file, and field-separator character can be specified by a prospector user. In Fig. 2, we can see an example of how to use the improved utility on the data presented in Fig. (1.b). The input file named Owner-Car.csv is preceded by the *-i* switch. Notice that the temporary file (preceded by the *-t* switch) is needed to redirect the original output of the miner and the output file (preceded by the *-o* switch) is only useful to a client/user. Additionally, the *-s* switch allows to specify a separator character for fields (viz., either a semicolon or a comma).

The functional dependencies found with the miner are presented in Fig. 3. We call these dependencies, *tentative functional dependencies*. Notice that some of these functional dependencies are the true ones (e.g. *IDPerson* → *Name*) but other ones are just contingent on the values present in the data (e.g. *Name* → *IDPerson*). The revision of these dependencies to discard the false ones from the set of tentative functional dependencies is the subject that the module that we have added to GF deals with, and that is the matter of the next section.

```
TANE-Sergio.exe -i "Owner-Car.csv" -o "result-owner-car.txt" -t
                "temporal-owner-car.txt" -s ";"
```

Fig.2. Improved interface for the TANE miner

```
            IDPerson → Name
IDPersonCarPerception → IDCar
IDPersonCarPerception → CarBrand
                Name → IDPerson
    NameCarPerception → IDCar
    NameCarPerception → CarBrand
               IDCar → IDPerson
               IDCar → Name
               IDCar → CarBrand
               IDCar → CarPerception
           CarBrand → IDPerson
           CarBrand → Name
           CarBrand → IDCar
           CarBrand → CarPerception
```

Fig.3. Functional dependencies found by the TANE miner in the *CarOwner* spreadsheet

4. Revising Functional Dependencies

Here, we present the new module in GF that allows a user to interactively revise functional dependencies computed from a CSV datasource. In Fig. 4, we present the user interface for using the new module.

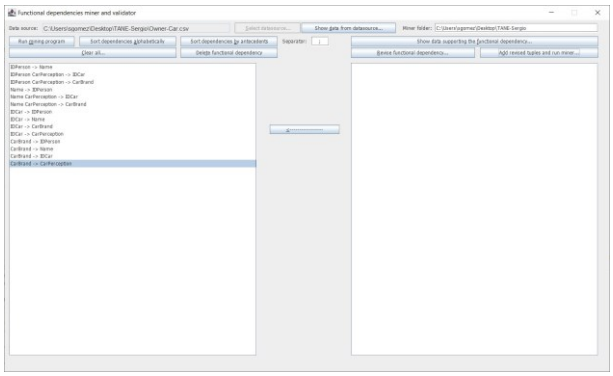


Fig.4. Module for mining and validating functional dependencies in CSV files

The controls in the GUI are enabled in a strict order that allows the user to unlock functionalities when input data is available for using them. This order is specified by the state diagram in Fig. 5. The usual workflow a user would follow when using the application will be: (1) opening the form takes him to state q_0; (2) specifying the datasource by selecting the particular CSV file he wants to work with (thus going into state q_1); (3) at any time the data from the data source can be explored; (4) running the dataminer as the one described in Sect. 3 computes the functional dependencies that are shown on the list on the left (notice that the miner program can be changed for an alternative program provided that it satisfies the interface as explained in Fig. 2), this will take the user to state q_2; (5) once the list is filled with the functional dependencies computed from the current data in the CSV, which we chose to call *tentative functional dependencies*, as they are contingent on current data, the user can select one of them getting to state q_3; (6) once in q_3, the user can optionally chose directly delete a tentative functional dependency or else to see the projection of the table limited to the columns mentioned in the selected functional dependency to determine if some combination of values is missing, determining so by pressing the button labeled “Revise FD” will take him to state q_4; (7) in q_4 the application will open a new form showing to the user invented tuples that are not currently in the table by producing random combinations with data already present in the table according to the algorithm in Fig. 7; (8) in q_5, the user can select one or more than one alternative tuples to add to the already existing tuples in the projected table, and then go to q_6; (9) in q_6, the miner is run on the projected table to validate if the revised dependency still holds or new ones appear, this new computed dependencies are displayed on the list located on the right, leading the system to state q_7; (10) in q_7, the user can delete the revised funtional dependency or select some of the newly found functional dependencies and copy them to the list on the left pane, and, (11) finally, the user can clear the process to return to state q_0.

Regarding step (2), our approach assumes that all fields are considered string typed, that no field delimiters are present and the user can choose only between comma and semicolon characters as field separators. Step (4) was already discussed in Sect. 3. To illustrate how steps (5)–(10) are intended to be used, we continue the preceding example. Consider the tentative functional dependency *CarBrand* → *CarPerception* computed in (4). The projection of the tuples restricted to its fields, and without duplicate data, is shown in Fig. (6.a). For example, suppose that the user considers that

the perception of *Chevrolet* must be *high* instead of *low*. The user can ask the system to generate unseen tuples in steps (6)-(7). This new invented tuples are generated using the algorithm in Fig. 7. Once the user explores the invented tuples, he can select the appropriate ones to be added to the projection (see Fig. 6.b). Then, after running again the miner, in this particular case, there are no functional dependencies that hold in this view of the table with the invented record that were selected. So the user can decide to delete the functional dependency under consideration. Conversely, as the system generates those potential scenarios, it lets the user deciding to keep or discard the proposed functional dependency. When the user observes that a suggested combination of data from the domain that respects the functional dependency is not consistent with his view of the domain, he can opt for discarding the suggested dependency. Besides, if other functional dependencies were discovered to hold in this new dataset, they will be shown in the list of right side of the screen, and then the user can decide to add to the main set of tentative functional dependencies.

5. Discussion and Related Work

Cunha et al. [3] present techniques and tools to transform spreadsheets into relational databases and back. A set of data refinement rules is introduced to map a tabular datatype into a relational database schema. Having expressed the transformation of the two data models as data refinements, they obtain for free the functions that migrate the data usin well-known relational database techniques to optimize and query the data. Because data refinements define bi-directional transformations, they can map such database back to an optimized spreadsheet. They implemented the data refinement rules and constructed Haskell-based tools to manipulate, optimize and refactor Excel-like spreadsheets. In our prototypical approach deals with Excel files but expressed as CSV tables, GF has the functionality to load Excel files and this feature can be added in the future. In our application, CSV files are handled internally as H2 tables but the data miner that we use as an external tool handles it as plain text. The work of Cunha et al. uses the set of functional dependencies to generate a normalized relational

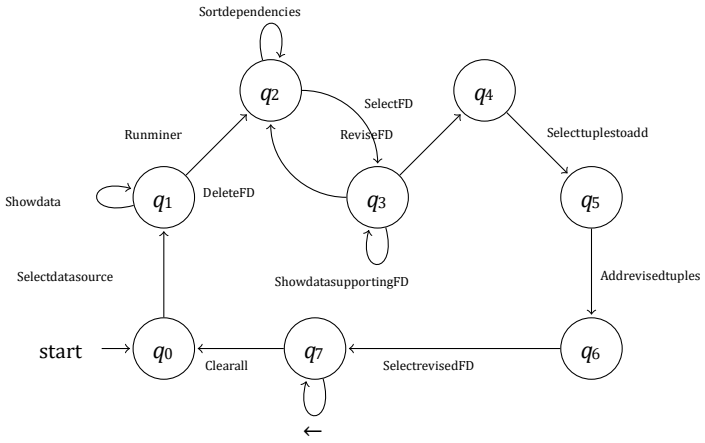
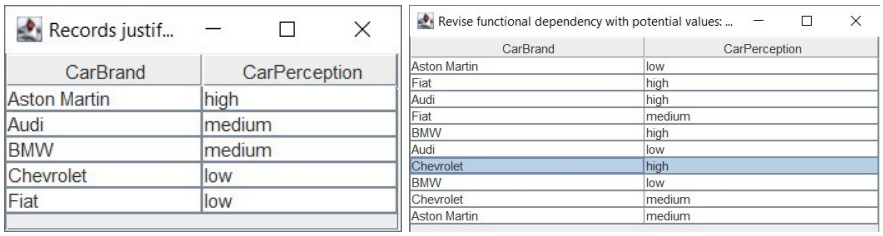


Fig.5. State diagram describing the usage of the GUI controls



(a) (b)

Fig.6. (a) Records justifying functional dependency *CarBrand* → *CarPerception* and (b) Invented records with alternative values

```
Algorithm GenerateAlternatives(Table T, FunctionalDependency C1...Cn → D)
Let values be a (n + 1)-size vector of sets
for i := 1 to n do
   valuesi := executeQuery(SELECT DISTINCT Ci FROM T)
valuesn+1 := executeQuery(SELECT DISTINCT D FROM T)
Let script be an empty sequence of strings
script.addLine(CREATE TABLE aux(C1 VARCHAR(100), ..., Cn VARCHAR(100), D
VARCHAR(100));)
for epoch := 1 to MAX TUPLES do
   Let x be a vector of n + 1 components
   for i := 1 to n + 1 do xi := randomElementFrom(valuesi)
   if x is a previously unseen combination of values then
      result := executeQuery(SELECT COUNT (*) AS Result FROM T
         WHERE C1 = x1 AND ...AND Cn = xn AND D = xn+1)
      if result = 0 then
         script.addLine(INSERT INTO aux(C1, ..., Cn, D) VALUES (x1, ...,
         xn+1);)
ExecuteSQLScriptToGenerateTableAndShow(script)
```

Fig.7. Algorithm for generating unknown alternatives from table T w.r.t. functional dependency $C_1 ... C_n \rightarrow D$

database from the contents of the spreadsheet. Our application does not do that yet and adding that functionality remains a future work. They use the FUN mining technique and we use the TANE mining technique. As our approach is fully customizable, we could employ the FUN miner in the future provided a functioning version of it is available. In particular, Cunha et al. present an example that considers a spreadsheet for representing a property renting system (see [3, Fig. 1]). In the accompanying files published online with our executable application, we provide a recreation of that file, named book.csv, showing that our system can compute the functional dependencies shown in that work.

Despite the ubiquity of spreadsheets, the problem of dealing with transformations of spreadsheet data to more formal data formats is still relevant. Müller and Mertov´a [4] justify and propose a lightweight recording-based solution to the tracing of the steps for transforming spreadsheets into ontologies that works on a wide variety of spreadsheet programs, from Microsoft Excel to Google Docs. GF can transform spreadsheets into ontologies but in this particular work we are concentrated in functional dependencies emerging from tables within spreadsheet data.

Salem and Abdo [5] propose two techniques for mining accurate conditional functional dependencies rules from such databases to be employed for data cleaning. The idea of the proposed techniques is to mine firstly maximal closed frequent patterns, then mine the dependable conditional functional dependencies rules with the help of lift measure. That approach is complementary to ours because in the current status of our work, we assume that the data is accurate. Their approach could be used to extend our solution with a data-cleaning preprocess for capturing semantic errors.

6. Conclusions and Future Work

In this paper we tackled the problem of finding functional dependencies in a spreadsheet table. We relied on the TANE datamining algorithm integrating it with our GF framework for ontology-based data access. We presented a running example that showed how our approach assists the user so he can revise a tentative set of functional dependencies computed with TANE using possible tuples not already present in the table that are computed by our system. The limitations of our approach are all fields are considered string typed, that no field delimiters are present and the used can choose only between comma and semicolon characters as field separators. The solution is customizable provided that a miner is provided satisfying a very precise command-line interface.

Possible research avenues for extending the work presented here include generating a normalized data base using the functional dependencies computed by the miner and later revised by the user with the assistance of our system, and then exporting this database as an ontology. Other possible paths to be explored are the adaptation of the mining algorithms to different data types and constraints.

Acknowledgments. This work was supported by Secretaría General de Ciencia y Técnica, Universidad Nacional del Sur, Argentina, and by Comisión de Investigaciones Científicas de la Provincia de Buenos Aires (CIC-PBA).

References

1. Huhtala, Y., Kärkkäinen, J., Porkka, P., Toivonen, H.: TANE–An efficient algorithm for discovering functional and approximate dependencies. The Computer Journal **42**(2) (1999) 100–111
2. Gómez, S.A., Fillottrani, P.R.: A Query-By-Example Approach to Compose SPARQL Queries in the GF Framework for Ontology-Based Data cmdAccess. In Pesado, P., ed.: 28th Argentine Congress, CACIC 2022 – Revised Selected Papers. Springer (2023) 211–226
3. Cunha, J., Saraiva, J., Visser, J.: From spreadsheets to relational databases and back. In: PEPM '09: Proceedings of the 2009 ACM SIGPLAN workshop on Partial evaluation and program manipulation. ACM (jan 2009) 179–188
4. Müller, W., Mertová, L.: ReStoRunT: Simple Recording, Storing, Running and Tracing changes in Spreadsheets. In et al., B.K.R., ed.: BTW 2023, Lecture Notes in Informatics (LNI), Gesellschaft für Informatik. LNI (2023) 865–877
5. Salem, R., Abdo, A.: Fixing rules for data cleaning based on conditional functional dependency. Future Computing and Informatics Journal **1**(1) (2016) 10–26

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Proyecto Aplicación para la Gestión de Residuos Sólidos Urbanos

Lic. Diego Martín Luna[1], Ing. Franco Pablo Quintero[1],
Mg. Rodolfo Bertone[1,2], C.C. Marcelo Taruschio[1,3]

[1] Profesor FACEI UCALP; [2] Secretario Académico UCAP; [3] Director de Carrera UCALP
{diegomartin.luna,francopablo.quintero}@ucalpvirtual.edu.ar,
{rodolfo.bertone, marcelo.taruschio}@ucalp.edu.ar

Abstract. El caso de estudio aborda la problemática de la falta de recolección y disposición adecuada de Residuos Sólidos Urbanos (RSU) en espacios públicos, lo que conlleva a la proliferación de roedores, insectos y animales, acentuando su dispersión y deterioro. La dispersión de los desechos es favorecida tanto por la actividad animal como por el viento, creando una coyuntura donde los residuos pueden ser arrastrados por el agua de lluvia hacia sistemas de drenaje, y en ciudades costeras, hacia el mar. Esta situación implica riesgos al propagarse microplásticos a través de la red trófica y alcanzar al ser humano, además de riesgos bacteriológicos significativos para la salud pública. El estudio se enfocará en mejorar la eficiencia en la fase de Recolección y Transporte de RSU, con el objetivo de disminuir el tiempo que los residuos permanecen en la vía pública, mediante una mejor coordinación entre los habitantes y el servicio de recolección local.

Keywords: Residuos Sólidos Urbanos (RSU). Geolocalización. Impacto ambiental. Microplásticos. Red trófica. Metodología Agile. Aplicación web móvil.

1. Introducción

Residuo es todo elemento, material, objeto o sustancia que, como consecuencia de los procesos de consumo y desarrollo de actividades humanas, es desechado o abandonado. Los residuos sólidos urbanos (RSU) son análogos a los denominados domiciliarios y pueden ser de origen residencial, urbano, comercial, asistencial, sanitario, industrial o institucional, con excepción de aquellos que se encuentren regulados por normas específicas.

La cantidad de RSU que generan (directa o indirectamente) los habitantes de un área refleja las condiciones de producción y consumo de la sociedad. Los residuos repercuten a largo plazo en la salud humana y el ambiente.

Cuando las instalaciones de recuperación y disposición final de RSU no cumplen con los requisitos mínimos de impermeabilidad de los suelos donde se emplazan, ni con las distancias a las napas freáticas, a los cursos de aguas superficiales, a los centros urbanos u otras áreas susceptibles de recibir los impactos derivados de estas instalaciones, se producen impactos negativos sobre el ambiente y la salud humana

En Argentina, el manejo de los RSU está regulado por la Ley de Presupuestos Mínimos 25916 que establece los presupuestos mínimos para un manejo adecuado de

los residuos domiciliarios, a partir de propender a una gestión integral de los mismos, propiciar su valorización y promover su minimización en la generación y disposición final. En términos generales, comprenden desechos de hogares y centros comerciales, oficinas e industrias que, dada su composición, son comparables con aquellos generados en domicilios particulares.

El 54 % de la población recibe el servicio de recolección en forma tercerizada y el restante 46 % como prestación municipal directa. La frecuencia de recolección diaria es superior al 70 %.

La cobertura de disposición final en RS del 64,7 % de la población esconde inequidades geográficas. Esta cobertura es menor en las regiones Norte (50,1 %) y Cuyo-Mesopotamia (15,2 %), siendo que en el resto del país es de 79,4 %.

El restante 35,3 % de la población cuenta con una disposición final inadecuada: 9,9 % en vertederos controlados, 24,6 % en basurales a cielo abierto.

La disposición final del 45 % de la población es atendida mediante servicio municipal directo, contratos de servicios que cubren al 24 % y otras modalidades que cubren al 31 %.

En las principales áreas metropolitanas se cuenta con plantas de separación, y se está desarrollando una industria para el procesamiento de los residuos recuperados (plásticos, vidrios, papel y cartón), los cuales son mayormente recolectados por recuperadores y recuperadoras, tanto en la vía pública como en sitios de disposición final. [1]

La falta de recolección y/o disposición adecuada de Residuos Sólidos Urbanos (RSU) de forma indefinida en espacios públicos conduce a la proliferación de roedores e insectos, lo cual también puede atraer a animales quienes pueden contribuir a su dispersión y deterioro. La dispersión de los residuos puede ser favorecida tanto por el viento como por la actividad de los animales mencionados. Además, esta situación implica riesgos bacteriológicos significativos para la salud y bienestar de la población en general.

La dispersión de los desechos en las vías públicas, mediada por el efecto del viento y la actividad animal, crea una coyuntura en la que se abre la posibilidad de que dichos residuos sean arrastrados por el agua de lluvia hacia los sistemas de drenaje de las ciudades. Esta situación conlleva a la conducción de los residuos hacia los cursos de agua, tales como ríos y arroyos, y en el caso de ciudades costeras, directamente hacia el mar.

Uno de los problemas ambientales globales que ha experimentado un deterioro significativo en los últimos años es la contaminación marina por residuos plásticos. Esta situación ha dado lugar a graves consecuencias que repercuten en múltiples niveles. Entre ellas, se destaca el daño causado a especies marinas y a hábitats de gran valor ecológico. La presencia de residuos plásticos en los océanos afecta a diversas criaturas marinas, desde peces hasta aves y mamíferos marinos, poniendo en peligro su supervivencia y, al propagarse a través de la red trófica y alcanzar al ser humano, amenaza nuestra propia salud.[2]

2. Caso de Estudio

La Gestión Integral de los Residuos Sólidos Urbanos abarca las fases de Generación, Recolección y Transporte, Tratamiento / Recuperación, Disposición Final, y/o Reinserción en el Mercado o Industria / Reciclado.[3]

El Proyecto de Investigación tiene como objetivo estudiar la fase de Recolección y Transporte, para mejorar la eficiencia en la coordinación entre los habitantes y el servicio de recolección. Puntualmente se centrará en la disminución del tiempo de permanencia de los residuos alocados en la vía pública. Se analizó el estado del arte en el tema. [4][5][6][7]

El objetivo es disminuir el intervalo de tiempo que transcurre desde que el vecino deposita sus residuos hasta que son recogidos por el servicio local. El gráfico 1, muestra las fases indicando el foco del caso de estudio.

2.1 Análisis del caso de estudio

El caso de estudio propuesto busca:

- Fomentar la conciencia y fomentar el compromiso con el medio ambiente, incentivando la participación activa de la comunidad en prácticas sostenibles de recolección de residuos urbanos.
- Emplear estratégicamente los recursos disponibles, considerando las restricciones económicas vigentes, con el propósito de optimizar las operaciones de recolección de residuos.
- Destacar y abordar una problemática en constante crecimiento relacionada con la gestión de residuos urbanos, brindando una mayor visibilidad a los desafíos ambientales que enfrenta la sociedad.
- Contribuir a mejorar la calidad de vida y el bienestar de los ciudadanos mediante la implementación de soluciones eficientes de recolección de residuos, que impacten positivamente en el entorno urbano y en la salud pública.



Gráfico 1: Fases de Tratamiento de residuo

3. Solución propuesta

Para mejorar la eficiencia en la coordinación entre los habitantes y el servicio de recolección, se propone investigar y desarrollar un software que permita la sincronización entre la recolección y el vecino que genera el residuo.

3.1 Escenarios

Los casos identificados a los que la comunidad se enfrenta al momento de alocar los residuos en la vía pública son:

- Sacar los residuos con demasiada anticipación al horario de recolección.
- Sacar los residuos posteriormente al paso del servicio de recolección.
- Sacar los residuos con el servicio de recolección suspendido.
- Sacar los residuos con el servicio de recolección demorado.
- Controlar la eficiencia en el servicio de recolección.

3.2 Historias de Usuarios

La tabla 1 define las historias de usuario que conforman el problema. Asimismo, se plantea la complejidad de las mismas y su prioridad para su tratamiento.

Historias de Usuario	Complejidad	Prioridad
HU01 – Gestión de unidad de recolección.	Baja/Media	Alta
HU02 – Gestión de posta de recolección.	Baja/Media	Alta
HU03 – Gestión de zonas y barrios.	Media	Media
HU04 – Gestión de tipos de residuos.	Baja/Media	Baja
HU05 – Gestión de días y horarios.	Media	Media
HU06 – Gestión de recorrido.	Media/Alta	Alta
HU07 – Consulta de recorrido.	Alta	Alta
HU08 – Gestión de favoritos.	Media	Baja
HU09 – Gestión de Incidentes.	Alta	Baja

Tabla 1: Historias de Usuario

3.3 Esquema de solución de software

La solución propuesta busca sincronizar a las partes con una aplicación compuesta por tres artefactos de software, el primero dedicado a la identificación y geolocalización en línea de la unidad de recolección, el segundo con el objetivo de informar al habitante el estado del servicio, el tiempo de llegada de la unidad de recolección a su posta, o informar que el servició ya pasó por la posta consultada, el tercero, un sistema de gestión y administración que articula los datos provenientes del primer artefacto, los procesa y los coloca en disponibilidad para los vecinos. El gráfico 2 presenta en forma esquemática la solución analizada.

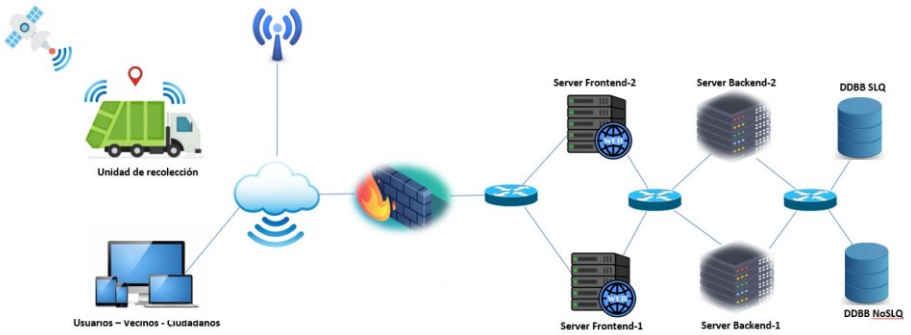


Gráfico 2: esquema de solución propuesta

3.4 Aplicaciones de terceros utilizadas

Con el propósito de garantizar el éxito del proyecto se aplica la disciplina de Ingeniería de Software [13], utilizando metodologías, herramientas de soporte y

tecnologías idóneas para llevar a cabo el proyecto. De esta manera, el objetivo es implementar el software asegurando su funcionalidad y escalabilidad.

3.4.1 Herramientas

El desarrollo exitoso de un proyecto de software requiere una gestión meticulosa y una coordinación eficiente entre los equipos de trabajo. En este sentido, las herramientas seleccionadas para brindar soporte en cada etapa del ciclo de vida del proyecto van desde plataformas de gestión de proyectos que permiten una planificación estructurada y seguimiento de tareas, hasta sistemas de control de versiones que facilitan la colaboración y el mantenimiento del código. Las herramientas elegidas son: Trello,

UML 2, Lucid chart, Slack, Git-GitHub. [8][9][10][11][12]

3.4.2 Tecnologías

La elección de las tecnologías adecuadas es un aspecto crítico en el desarrollo de software (se procura el uso de software libre), ya que puede influir en el rendimiento, la seguridad y la experiencia del usuario final. En esta sección, se expondrán las tecnologías seleccionadas para la implementación del software en cuestión. Estas son: OpenJDK, Debian, Java, Payara Server, Docker, PostgreSQL, MongoDB Atlas.

3.5 Estrategia

Siguiendo la naturaleza del desarrollo de un producto de software, la estrategia se forja a partir de la identificación de un mapa de impacto, que sienta las bases de las necesidades de los requerimientos y expectativas de los usuarios. A partir de las necesidades se define un *roadmap* de historias de usuarios hacia el logro del producto deseado.

A partir de las historias de usuarios identificadas se crea el *release plan* conjuntamente con su MVP (Mínimo Producto Viable) como punto de partida y se empleará una metodología Agile, conocida por su carácter iterativo e incremental, para avanzar sobre el análisis, diseño, desarrollo, pruebas y puesta en producción de forma adaptativa y ágil, permitiendo así una evolución continua y una entrega temprana de valor al cliente.

3.6 Gestión de Riesgo

La gestión de riesgos en el proyecto está enfocada en la geolocalización en línea de las unidades de traslado, la identificación de puntos o postas de recolección y el armado de los recorridos de recolección. Estas tres funcionalidades se presentan como un componente esencial para asegurar el éxito y la efectividad del proyecto. Entre los riesgos potenciales a considerar se encuentran la falta de precisión en la geolocalización, problemas de conectividad en tiempo real, vulnerabilidades de

seguridad en la transmisión de datos y posibles fallos técnicos que puedan afectar el rendimiento del sistema.

3.6.1 Tratamiento de riesgos y mitigación:

Falta de precisión en la geolocalización: Realizar pruebas exhaustivas de geolocalización en diferentes entornos y condiciones, utilizando distintos dispositivos y redes. Implementar algoritmos de corrección y calibración para mejorar la precisión de las coordenadas obtenidas. Establecer un sistema de feedback y retroalimentación de usuarios para recopilar información sobre posibles inexactitudes y ajustar el software de acuerdo con sus comentarios.

Problemas de conectividad en tiempo real: Diseñar el sistema para que pueda operar de manera óptima en situaciones de baja o intermitente conectividad. Implementar técnicas de almacenamiento en caché para permitir que la aplicación funcione en modo offline temporalmente y sincronice los datos cuando la conexión se restablezca. Utilizar tecnologías y protocolos de comunicación robustos para garantizar la transmisión confiable de datos en tiempo real.

Vulnerabilidades de seguridad en la transmisión de datos: Aplicar prácticas de codificación segura y realizar pruebas de penetración para identificar y abordar posibles puntos débiles en el sistema. Implementar protocolos de cifrado y autenticación para proteger la confidencialidad e integridad de los datos transmitidos. Mantenerse actualizado con las mejores prácticas de seguridad y aplicar parches y actualizaciones de seguridad de forma regular.

Posibles fallos técnicos que afecten el rendimiento del sistema: Realizar pruebas de carga y rendimiento exhaustivas para identificar posibles cuellos de botella y puntos de falla en el sistema. Implementar mecanismos de monitoreo y registro de errores para detectar y resolver rápidamente problemas técnicos. Establecer un plan de contingencia para situaciones de emergencia y contar con respaldos y planes de recuperación ante posibles fallos.

Dificultades en la adopción del sistema por parte de los usuarios: Realizar una exhaustiva investigación de usuarios para entender sus necesidades y expectativas. Diseñar una interfaz de usuario intuitiva y fácil de usar, con una experiencia de usuario atractiva. Proporcionar un canal de comunicación abierto para recibir sus comentarios y sugerencias.

Datos de geolocalización inconsistentes o incompletos: Implementar controles de calidad para verificar la integridad de los datos de geolocalización antes de ser incorporados al grafo. Establecer procedimientos para identificar y corregir posibles errores o lagunas en la información geográfica recopilada. Realizar auditorías regulares de los datos para asegurar su validez y actualización.

Errores en el cálculo del recorrido a través del grafo conexo con peso: Verificar y validar la lógica y los algoritmos utilizados para calcular el recorrido en el grafo. Realizar pruebas exhaustivas de los algoritmos de ruta para asegurar su precisión y eficiencia. Implementar mecanismos de registro de errores y excepciones para identificar y resolver problemas en tiempo real.

3.7 Funcionamiento

La aplicación web diseñada para múltiples dispositivos (Smat Phone/Laptop/Tablet - *responsive design*) ofrece una experiencia interactiva y funcional, permitiendo a los usuarios/vecinos consultar en tiempo real la demora exacta de la unidad de recolección de residuos, en función de la posta de recolección seleccionada. Esta característica facilita la coordinación y la puntualidad en los encuentros, ya que los usuarios/vecinos pueden monitorear la aproximación de la unidad de recolección en tiempo real, lo que les permite organizarse con tranquilidad para depositar los residuos en la vía pública.

Al acceder a la aplicación, los usuarios podrán establecer puntos de recolección favoritos, asimismo, la aplicación informa el estado actual del servicio, demorado (indicando el tiempo de demora), suspendido o cambio de recorrido, exponiendo un mensaje en pantalla que indica que "***por favor retire el residuo de la posta o evite sacarlo***". El gráfico 3 muestra el esquema de funcionamiento previsto.

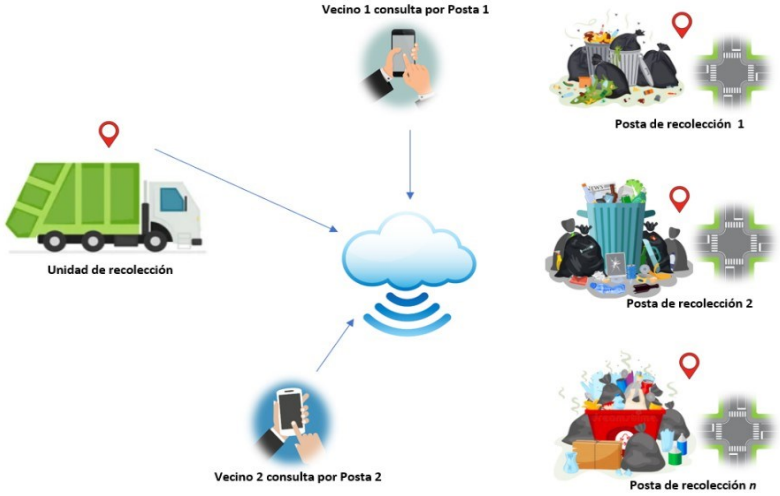


Gráfico 3: Esquema de funcionamiento

4. Resultados Esperados

El presente estudio tiene como objetivo desarrollar una aplicación innovadora para controlar el movimiento de los camiones de residuos y mejorar la eficiencia del servicio de recolección de basura en entornos urbanos. La propuesta se centra en utilizar tecnologías de seguimiento en tiempo real para proporcionar a los usuarios información actualizada sobre la proximidad de los camiones de recolección, lo que les permitirá optimizar el momento adecuado para desechar sus residuos. Se espera que la aplicación contribuya a reducir el tiempo de los residuos en la vía pública, dar certeza sobre la espera del servicio para los usuarios y, por ende, disminuir la probabilidad de dispersión de los desechos en las calles, abordando así problemas ya mencionados sobre la contaminación ambiental.

Además del control del movimiento de los camiones de residuos, el estudio busca implementar un sistema de monitoreo y evaluación del servicio de recolección. Mediante el análisis de datos recopilados por la aplicación, se pretende identificar posibles ineficiencias en las rutas de recolección y proponer ajustes estratégicos para optimizar los recorridos. Se espera que esto mejore los indicadores clave de desempeño, como la puntualidad en la recolección y el tiempo promedio de recolección por ruta, lo que llevará a una prestación de servicio más eficiente y efectiva para los ciudadanos.

En línea con el objetivo de mejorar la experiencia del usuario, la aplicación permitirá una interacción directa entre los usuarios y el servicio de recolección. Se implementará una funcionalidad para que los usuarios puedan reportar incidencias o problemas relacionados con la recolección de residuos de manera rápida y sencilla. Se espera que este sistema de retroalimentación instantánea facilite la resolución de quejas y contribuya a una mayor satisfacción del cliente con el servicio.

En cuanto a la sostenibilidad ambiental, se espera que la aplicación fomente práctica más responsables en la gestión de residuos. Al proporcionar información transparente sobre la recolección de basura y la proximidad de los camiones en tiempo real, se busca concientizar a la población sobre la importancia de una correcta disposición de los desechos y promover la participación en acciones de reciclaje y reducción de desperdicios. Se prevé que esta solución tenga un impacto positivo en el entorno urbano y contribuya a avanzar hacia un modelo más sostenible de recolección de residuos. Para esto, el grupo de trabajo pretende incorporar docentes investigadores y alumnos avanzados de la carrera de Ingeniería Ambiental, que se dicta también en la institución.

5. Conclusiones y Trabajos Futuros

El desarrollo de una aplicación para controlar el movimiento de camiones de residuos y mejorar el servicio de recolección, ha demostrado ser una iniciativa prometedora para abordar los desafíos asociados con la gestión de residuos urbanos. Las expectativas son altas en cuanto al potencial de la aplicación para optimizar el proceso de recolección, reducir los tiempos de espera para los usuarios y mejorar la calidad de vida en entornos urbanos, al minimizar el impacto ambiental de la acumulación de basura en las calles.

La propuesta de seguimiento en tiempo real de los camiones de recolección y la interacción directa con los usuarios a través de la aplicación permitirían una mayor transparencia en el servicio, lo que fomentaría la confianza y participación activa de la comunidad en prácticas más responsables de disposición de residuos. Además, el monitoreo y análisis de datos proporcionados por la aplicación facilitan una gestión más eficiente y efectiva del servicio, lo que podría traducirse en una disminución de los costos operativos y una optimización de los recursos disponibles.

Se considera aún que hay aspectos que requieren atención en trabajos futuros para mejorar y consolidar la aplicación propuesta:

- Evaluación y validación empírica: Es fundamental llevar a cabo un estudio piloto e implementar la aplicación en un entorno real para evaluar su efectividad y recopilar datos reales que respalden las conclusiones teóricas. La validación empírica permitiría ajustar y mejorar la aplicación en función de la

retroalimentación de los usuarios y los resultados obtenidos.

- Integración con sistemas existentes: Considerar la integración de la aplicación propuesta con sistemas de recolección de residuos ya establecidos en las ciudades. La compatibilidad con plataformas y tecnologías existentes facilitaría la adopción y aceptación por parte de los actores involucrados en la gestión de residuos.
- Análisis de impacto ambiental: Realizar un análisis de impacto ambiental para evaluar cómo la aplicación puede contribuir a reducir la huella de carbono y los problemas de contaminación asociados con la gestión de residuos. Esto permitiría cuantificar los beneficios ambientales y generar conciencia sobre la importancia de adoptar prácticas más sostenibles.

Referencias

[1] https://www.argentina.gob.ar/ambiente/control/rsu

[2] https://www.argentina.gob.ar/ambiente/basuramarina

[3] https://www.argentina.gob.ar/ambiente/control/rsu/etapas

[4] Optimization Model for Waste Collection: A Case Study in the City of Campinas, Brazil.Rodrigo C. M. Fernandes, Luiz Fernando M. Heineck, Luis F. Autran M. Pereira. 2017. WasteEng2016 - 6th International Conference on Engineering for Waste and Biomass Valorization.

[5] Smart Waste Management System for Efficient and Sustainable Waste Collection.S. R. Sawant, S. N. Mali, A. R. Surve, D. S. Bormane. 2018. 2018 International Conference on Information Technology (ICIT).

[6] An IoT-based Smart Waste Management System for Efficient and Sustainable Waste Collection. Amit Kumar Dutta, Harshit Srivastava, Rajiv Kumar, Anuj Kumar Dwivedi, Bhupendra Verma. 2019. 2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS).

[7] A Review of Smart Waste Collection Technologies towards Sustainable Smart Cities. Rishi Sharma, Shikha Sharma. 2019. 2019 International Conference on Smart Systems and Inventive Technology (ICSSIT).

[8] Trello: A System for Managing Software Projects.Michael G. McNeill, Ian Gorton, Tom Anderson, Tim Pfeil, Christian Roth, Brandon Lloyd, Jon Reeves, Shawn Blanton, Adam Porter.2008 IEEE International Conference on Software Maintenance (ICSM).

[9] The Unified Modeling Language User Guide (2nd Edition)Grady Booch, James Rumbaugh, Ivar Jacobson. Año: 2005.Editorial: Addison-Wesley Professional.

[10] Collaborative Drawing Tools for the Classroom. Lucid Software Inc.2014 American Society for Engineering Education Annual Conference & Exposition (ASEE). 2014. - URL: https://peer.asee.org/collaborative-drawing-tools-for-the- classroom

[11] Intelligent Messaging for Teams.Stewart Butterfield, Eric Costello, Cal Henderson, Serguei Mourachov. 2016 ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW).

[12] Título: "GitHub: A Web-Based Hosting Service for Version Control Repositories. Tom Preston-Werner, Chris Wanstrath, PJ Hyett. 2009. - URL: https://scholarworks.umass.edu/cgi/viewcontent.cgi?article=1169&context=csfaculty pubs

[13] La ingeniería de software es una disciplina de la ingeniería que se interesa por todos los aspectos de la producción de software. Ingeniería de Software. Sommerville, Ian. Novena edición Pearson Educación, México, 2011

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Digitalización 3D para aplicaciones móviles con realidad virtual y realidad aumentada

Agustín Cao, Sebastián Dapoto[0000-0001-7593-0198], Pablo Thomas[0000-0001-9861-987X], Patricia Pesado[0000-0003-0000-3482]

Instituto de Investigación en Informática LIDI, Facultad de Informática, Universidad Nacional de La Plata, La Plata, Argentina alcaolpg@alu.ing.unlp.edu.ar {sdapoto, pthomas, pesado}@lidi.info.unlp.edu.ar

Resumen. Las aplicaciones móviles 3D, en especial aquellas que incluyen funcionalidad de realidad virtual o realidad aumentada, han revolucionado la forma en que las personas interactúan con el mundo digital. Para poder crear los escenarios 3D de estas aplicaciones, se precisa contar con los modelos necesarios, y muchas veces estos modelos no se encuentran a disposición. Este trabajo muestra las tecnologías existentes para llevar a cabo el proceso de digitalización 3D de objetos reales, y su posterior uso en el desarrollo de aplicaciones móviles 3D. En particular, se presenta como caso de estudio el proceso llevado a cabo en la creación de una aplicación móvil 3D con realidad aumentada para el estudio de instrumental quirúrgico para cirugía general veterinaria.

Palabras Clave: digitalización 3D, realidad virtual, realidad aumentada, unity, cirugía general veterinaria, instrumental quirúrgico, m-learning.

1. Introducción

Las aplicaciones móviles 3D se basan en la representación y manipulación de objetos tridimensionales en un entorno virtual. Utilizan técnicas y algoritmos para crear y renderizar gráficos 3D.

Dentro del proceso de desarrollo de una aplicación móvil 3D se destaca el modelado 3D de los objetos que se visualizarán luego en la aplicación. Estos objetos pueden ser creados desde el mismo entorno de desarrollo de la aplicación o a través del uso de alguna herramienta externa y específica de modelado 3D.

Existen diversas técnicas para la creación de modelos 3D; una de las más tradicionales es el modelado mediante herramientas de software para gráficos 3D por computadora. De igual forma, existen hoy en día otras tecnologías que permiten lograr modelos 3D a partir de elementos físicos, como la fotogrametría, el escaneo 3D e incluso la generación de modelos mediante el uso de Inteligencia Artificial (IA). Una vez que se cuenta con el conjunto de objetos 3D a utilizar, es posible aplicar materiales o texturas a dichos objetos, de forma de lograr que se vean lo más realistas posible.

Particularmente, las aplicaciones móviles 3D aprovechan las capacidades táctiles y gestuales de los dispositivos móviles para permitir a los usuarios interactuar con objetos tridimensionales. A su vez, una aplicación 3D puede contener funcionalidad de realidad virtual (RV) o realidad aumentada (RA), haciendo que la forma de interacción con el usuario cambie completamente.

La realidad virtual posibilita al usuario sumergirse en escenarios tridimensionales en primera persona, a través del uso de cascos de RV, HMD (Head Mounted Displays), o incluso mediante dispositivos más simples, como Google Cardboard, que hacen uso de la pantalla dividida del smartphone para proyectar una imagen diferente para cada ojo. Las aplicaciones con RV permiten al usuario sentirse inmerso y presente en el entorno simulado. La interacción en las aplicaciones con RV puede lograrse a través de distintos tipos de dispositivos, como por ejemplo controles de mando manuales, guantes o trajes especiales, en conjunto con otros sensores externos como cámaras, haciendo posible la interacción sin el uso de dispositivos externos, mediante diferentes técnicas aplicadas a la detección de gestos y movimientos.

Por otro lado, la realidad aumentada combina elementos del mundo real con objetos virtuales. Las aplicaciones móviles 3D pueden utilizar tecnologías de RA para superponer objetos 3D en el entorno físico, permitiendo a los usuarios interactuar con ellos a través de la cámara de su dispositivo móvil. Ronald Azuma propuso en 1997 tres criterios o características que una aplicación de RA debería cumplir para considerarse como tal. [1]: Combinar elementos reales y virtuales; ser interactivo en tiempo real y registrarse en 3D.

La RA permite mejorar la interacción de un usuario con el mundo real y la percepción que el usuario tiene sobre el entorno. Por lo tanto, se vale de dispositivos como cámaras, giroscopios y módulos de geolocalización o GPS, que permiten realizar operaciones como seguimiento de imágenes, de movimiento, de rostros y de objetos, permitiendo realizar un análisis del entorno.

El presente trabajo muestra las técnicas existentes para llevar a cabo el proceso de digitalización 3D de objetos reales, y su posterior uso en el desarrollo de aplicaciones móviles 3D. Además, se presenta como caso de estudio el proceso de digitalización y desarrollo llevado a cabo en la creación de la aplicación móvil 3D con realidad aumentada denominada CGRA (Cirugía General en Realidad Aumentada). Esta aplicación sirve para el estudio de instrumental quirúrgico de cirugía general veterinaria, por lo que el proyecto fue llevado a cabo mediante un equipo multidisciplinario conformado por miembros del Instituto de Investigación en Informática LIDI de la Facultad de Informática, en colaboración con la cátedra de Cirugía General de la Facultad de Ciencias Veterinarias, ambas facultades pertenecientes a la Universidad Nacional de La Plata (UNLP).

Este trabajo se organiza del siguiente modo: a continuación, se mencionan los trabajos relacionados; luego se presentan las diferentes técnicas para llevar a cabo el proceso de digitalización 3D. En la siguiente sección se detalla el caso de estudio: la motivación del proyecto, el proceso de digitalización de las herramientas quirúrgicas y el desarrollo de la aplicación móvil. Seguido a esto se muestran los resultados finales obtenidos. Finalmente se presentan las conclusiones y trabajos futuros.

2. Trabajos relacionados

El modelado 3D se inserta en múltiples campos de manera amplia y en constante evolución. Por ejemplo, en el área enfocada al proceso educativo existen diversas aplicaciones móviles que complementan el estudio en diversos campos.

Teniendo en cuenta el caso particular planteado en este trabajo, las aplicaciones existentes destinadas al estudio de instrumental quirúrgico contienen tan solo imágenes y descripciones de las herramientas. Un ejemplo es la aplicación "General Surgery Instruments" [2], desarrollada por Two Arrows Development. Otro caso similar es el de "Surgical Instruments" [3], una aplicación paga disponible en español desarrollada por RER MedApps. Un ejemplo donde sí se utilizan objetos 3D es "AR Atom Visualizer" [4], una aplicación desarrollada por Signal Garden Research, que utiliza RA para crear proyecciones 3D animadas de átomos con una nube de electrones.

La aplicación CGRA [5] se piensa como un aporte y evolución para el proceso tradicional de enseñanza de las herramientas orientadas a cirugía general, el cual podría verse beneficiado mediante la incorporación de los modelos 3D de el instrumental quirúrgico y su visualización mediante RA.

Por otro lado, la división de Arqueología del Museo de La Plata, perteneciente a la Facultad de Ciencias Naturales de la UNLP, cuenta con investigadores trabajando con tecnología de modelado 3D, fotogrametría y escaneo 3D, con la finalidad de preservar y digitalizar artefactos arqueológicos como el representado en la figura 1, e incluso sitios de gran valor arqueológico para la humanidad, como lo es el Shincal de Quimivil, ubicado en la provincia de Catamarca, Argentina.



Fig. 1. Escaneo 3D realizado por el Museo de La Plata - Urotherium Antiquus [6].

3. Proceso de Digitalización

La digitalización 3D consiste en la recolección de parámetros e información de figuras y superficies del mundo real, para luego generar un modelo 3D a partir de los datos recuperados. Existen múltiples métodos por los cuales dicha información puede ser obtenida y procesada.

A continuación, se detallarán los principales métodos para adquirir los datos tridimensionales de un objeto: modelado 3D por computadora, escaneo 3D, fotogrametría y generación asistida por IA.

3.1 Modelado 3D por computadora

El software de modelado gráfico es un conjunto de herramientas utilizado con la finalidad de crear modelos 3D por computadora, los cuales pueden utilizarse para proyectos que incluyan animaciones, efectos visuales, aplicaciones 3D interactivas, entre otros.

Dentro del campo de los gráficos por computadora, el modelado 3D es el proceso de generar una representación matemática de una superficie u objeto en tres dimensiones mediante la utilización de software. Este proceso consiste en manipular vértices, bordes y superficies poligonales en un espacio tridimensional simulado, logrando así representar un objeto tridimensional utilizando un conjunto de vértices interconectados y formando figuras geométricas que conforman a los modelos. En la figura 2 es posible observar la manipulación de un modelo que representa una herramienta quirúrgica.

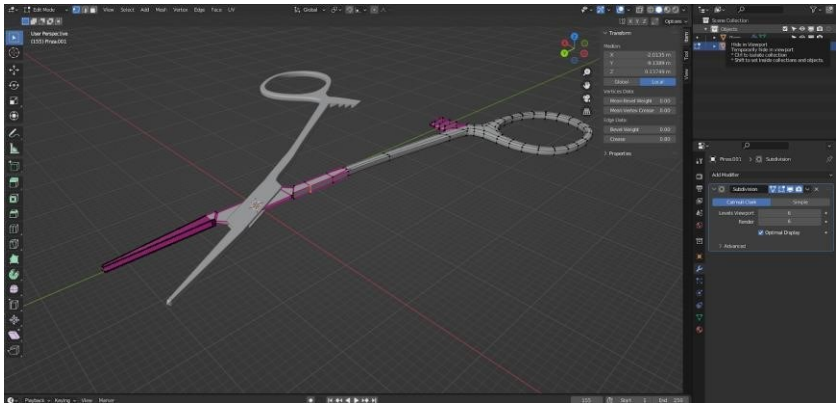


Fig. 2. Proceso de creación de una de las herramientas para CGRA mediante el software de modelado 3D.

Herramientas como Blender [7] son utilizadas para generar diversos modelos tridimensionales, entre otras funcionalidades como animaciones, texturas y mapas UV1. Mientras que Blender ofrece una experiencia enfocada en generar modelos

[1] Un mapa UV es una representación bidimensional de la superficie de un objeto tridimensional. Es una técnica utilizada para asignar texturas y detalles visuales a un modelo tridimensional.

orgánicos, otras herramientas, como FreeCAD [8], ofrecen una experiencia distinta de modelado, más orientada a la producción rápida de modelos para ingeniería.

3.2 Escaneo 3D

A diferencia del modelado 3D por computadora, existen métodos más automatizados para la generación de modelos tridimensionales. El escaneo 3D es uno de los métodos más utilizados hoy en día para la digitalización de objetos. Esta técnica se basa en la utilización de láser o luz infrarroja para crear un modelo digital de un objeto, midiendo la distancia al mismo y la variación de la luz al recorrer la superficie del objeto.

Esta técnica produce modelos con un gran nivel de detalle, por lo que es ideal para obtener información de geometrías complejas. Sin embargo, los dispositivos utilizados para esta técnica suelen ser sumamente costosos. En la figura 3 se puede observar un ejemplo de un dispositivo de escaneo 3D.



Fig. 3. Dispositivo de escaneo 3D.

3.3 Fotogrametría

La fotogrametría, por otro lado, utiliza imágenes fotográficas de un objeto obtenidas desde distintos ángulos y perspectivas para generar un modelo 3D de dicho objeto. La información se obtiene a partir del parallax[2] existente entre las imágenes tomadas al

[2] Parallax: diferencia o desplazamiento en la posición aparente de un objeto observado desde dos líneas de visión diferentes.

objeto. Aunque, en ciertos casos, los resultados no son tan precisos, pueden resultar satisfactorios a la hora de llevar objetos del mundo real a un entorno digital.

Mas allá de que esta técnica puede utilizarse en objetos relativamente pequeños como la estatua mostrada en la figura 4, el uso de la fotogrametría está de igual manera asociado al escaneo de estructuras grandes mediante el uso de drones equipados con cámaras y complementados con herramientas de sistemas de información geográfica.

La herramienta Meshroom es un software libre de reconstrucción 3D basado en el framework de fotogrametría AliceVision. [9]



Fig. 4. Estatua en proceso de digitalización, mediante la utilización de AliceVision.

3.4 Generación de modelos asistida por IA

Otro método que ha surgido en los últimos tiempos es el de generación de modelos tridimensionales asistido mediante inteligencia artificial. Este método opera a partir de una descripción textual o incluso usando como entrada una única imagen o fotografía de un objeto para generar rápidamente el modelo tridimensional solicitado. El proyecto Alpha3D [10] se enfoca en lograr un sistema preciso de modelado utilizando este método.

4. Caso de estudio: CGRA

4.1 Motivación

El proceso educativo tiene la necesidad de facilitar la construcción de conocimiento, y para lograrlo genera una búsqueda constante de innovación en los métodos de

enseñanza y la presentación de los contenidos. Durante la época de pandemia el acceso a materiales de estudio, como por ejemplo las herramientas físicas, se vio gravemente limitado. Esto, a su vez, profundizó la brecha existente entre aquellos alumnos que pueden adquirir los materiales y los que no.

CGRA es un proyecto que busca generar soluciones que acorten la brecha de conocimiento entre los alumnos, e incentive el estudio a partir de contenido didáctico y atractivo, utilizando tecnologías de realidad aumentada.

La aplicación móvil desarrollada asiste a los estudiantes de la asignatura Cirugía General de la Facultad de Ciencias Veterinarias de la UNLP en el estudio del instrumental quirúrgico utilizado en prácticas veterinarias. A través del uso de realidad aumentada, la aplicación permite visualizar modelos 3D en escala real del instrumental quirúrgico, pudiendo variar la orientación y el acercamiento, con el fin de poder visualizar los detalles de las herramientas.

4.2 Digitalización del instrumental

Para comenzar con el desarrollo de la aplicación móvil 3D con realidad aumentada, inicialmente fue necesario disponer de los modelos tridimensionales de las herramientas de cirugía a visualizar.

En un principio, se planteó la posibilidad de comprar los modelos terminados. Existen múltiples repositorios web que cuentan con modelos 3D de instrumental quirúrgico. Sin embargo, más allá del costo económico elevado que pueden conllevar ciertos modelos, al analizar dichos elementos en conjunto con los profesionales en el campo y bibliografía afín a la temática abarcada [11], se pudo llegar a la conclusión de que muchos de los modelos disponibles no cuentan con características clave propias del instrumental de interés. Por otro lado, entre el instrumental solicitado, existen piezas muy específicas que no se encuentran disponibles en dichos repositorios.

Entre los métodos anteriormente mencionados, se decidió comparar y utilizar dos de los métodos principales: fotogrametría y modelado 3D por computadora.

Respecto al método de fotogrametría, se realizó un estudio teórico junto con pruebas empíricas donde se sumaron profesionales expertos del Museo de La Plata. Sin embargo, luego de experimentar con diversos softwares como Meshroom y Metashape [12], específicos para esta tarea, además de utilizar múltiples configuraciones de los programas y variados procesos de fotografiado, se llegó a la conclusión de que la fotogrametría no sería la mejor forma de digitalizar el instrumental requerido. Las principales razones engloban el tiempo requerido para lograr obtener un modelo funcional, y la baja calidad en detalles puntuales que no llegó a satisfacer a los profesionales de la cátedra de Cirugía General de la Facultad de Ciencias Veterinarias de la UNLP.

El otro método, el modelado 3D mediante software específico, resultó ser la opción más adecuada para la tarea a realizar. Para dicho proceso, se decidió utilizar el software gratuito de código abierto Blender. Mediante este software no sólo se logró un nivel de detalle muy satisfactorio, sino que una vez conseguido el entrenamiento y la práctica necesarios para el uso de la herramienta [13][14], la creación de los modelos llevó en ciertos casos una fracción del tiempo requerido para lograr un resultado de calidad inferior mediante fotogrametría.

Entre las múltiples herramientas que provee Blender, existe una encargada de generar subdivisiones de forma rápida mediante el algoritmo Catmull-Clark[3]. Como puede observarse en la figura 5, las subdivisiones permiten llevar un modelo de bajos polígonos con pocos detalles, a un objeto con superficies altamente realistas sin demasiado trabajo. La ventaja se encuentra en la adaptación de la cantidad de polígonos generados de manera rápida. Un exceso en la cantidad de polígonos de un modelo puede llevar a un mal rendimiento gráfico de la aplicación, debido a la complejidad a la hora de procesar los objetos, afectando negativamente la experiencia del usuario.

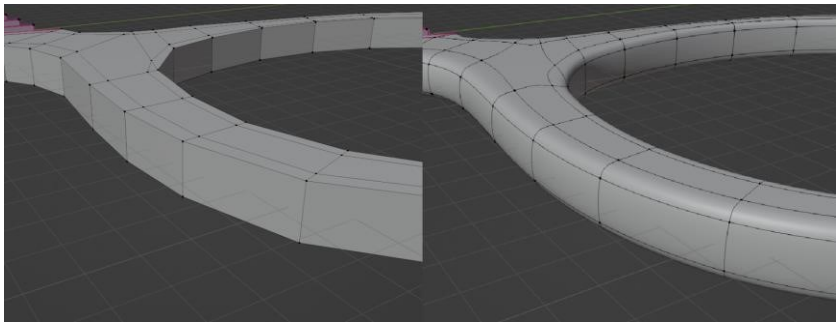


Fig. 5. A la izquierda, un anillo de una pieza sin subdivisiones; a la derecha, el mismo anillo con un modificador de subdivisión, aplicando subdivisión Catmull-Clark.

4.3 Desarrollo de la aplicación móvil

Una vez que se concluyó el diseño de una serie de modelos, y dichos modelos fueron aprobados por los profesionales docentes de la Facultad de Ciencias Veterinarias de la UNLP, se procedió con el diseño y posterior desarrollo de un prototipo de aplicación móvil 3D.

Para llevar a cabo esta tarea, se decidió utilizar la plataforma de desarrollo Unity [16]. Esta plataforma otorga las herramientas necesarias para el desarrollo de una aplicación móvil con RA mediante el framework AR Foundation [17], el cual engloba herramientas como ARCore [18], y las interfaces necesarias para crear un proyecto escalable y multiplataforma.

Unity además provee un extenso repositorio con herramientas que permiten desde generar animaciones hasta generar campos de texto con variadas configuraciones. Estos componentes luego pueden agregarse como atributos a los objetos que se representan en la aplicación. Además, para la primera etapa de desarrollo, se decidió centrar el proyecto en la plataforma Android, debido a su mayor alcance en la región y la simplicidad del desarrollo.

Se realizaron versiones de prueba con distintas características implementadas, que más adelante darían forma a la base de la aplicación. Por ejemplo, se hicieron pruebas sobre los posibles modos de visualización de las herramientas quirúrgicas en pantalla.

[3] El algoritmo de Catmull-Clark es una técnica utilizada en gráficos tridimensionales por computadora para crear superficies curvas a partir de la subdivisión de superficies. [15]

Estas versiones preliminares fueron evaluadas en conjunto con los responsables de la Cátedra de Cirugía General y un grupo de estudiantes, a fin de recolectar información que permitiera ajustar el prototipo y así llegar a su versión actual.

5. Resultados

En esta sección se muestra la aplicación denominada CGRA como resultado final del desarrollo propuesto en el presente trabajo.

A través del uso de marcadores QR, la aplicación permite acceder a representaciones de modelos 3D del instrumental quirúrgico en escala real. Es posible la visualización de las herramientas desde cualquier punto de vista. En la figura 6-a es posible observar que la herramienta aumentada respeta la escala de la herramienta real.

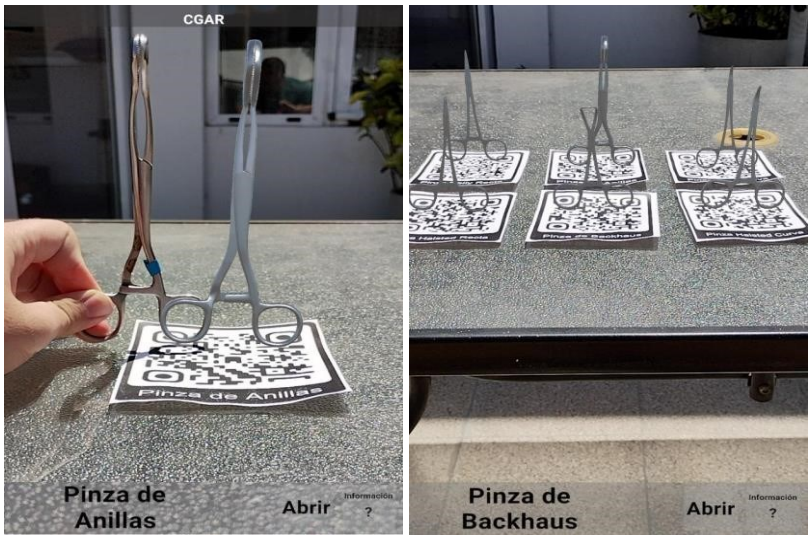
Por otro lado, es posible acercarse y alejarse del modelo aumentado, lo que permite la observación de los detalles físicos de las herramientas. Esto último es de gran importancia, ya que en ocasiones las diferencias entre una herramienta y otra pueden ser mínimas, y es necesario verlas desde muy cerca. Además, se brinda una descripción e información asociada a cada herramienta.

Otra funcionalidad implementada permite que una herramienta, por ejemplo, una pinza hemostática, abra y cierre sus ramas para observar detalles de su estructura y forma de uso.

En la figura 6-b es posible observar el conjunto de herramientas modeladas. En esta primera etapa del proyecto se resolvió la implementación de seis herramientas diferentes, típicas del set de instrumental quirúrgico básico visto por la cátedra Cirugía General de la Facultad de Ciencias Veterinarias de la UNLP y en la bibliografía asociada.

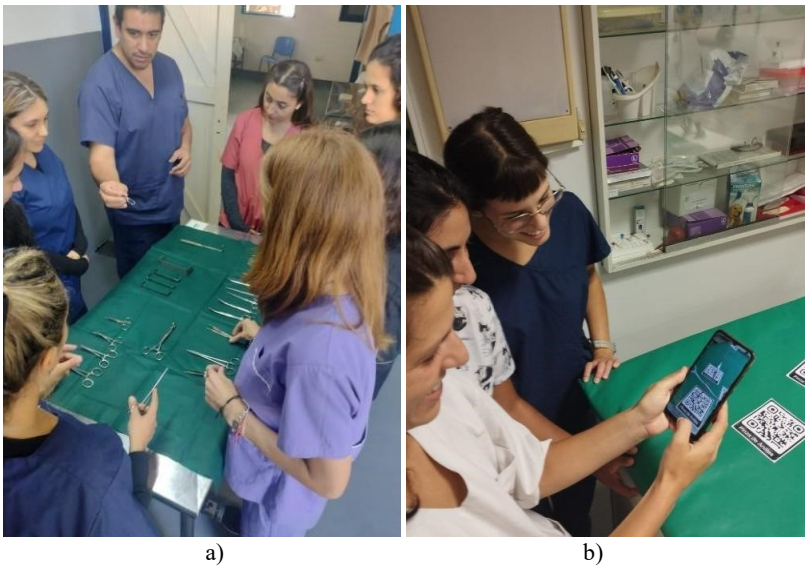
La forma tradicional de enseñar el uso del instrumental quirúrgico incluye la disponibilidad de uno o más sets reales de las herramientas y el traslado de éstos al aula. Los alumnos deben asistir a la clase de forma presencial y turnarse para poder tener en sus manos las herramientas y de esta forma poder apreciar los detalles de cada una de ellas. En la figura 7-a es posible ver a un grupo de alumnos observando el instrumental quirúrgico en clase. En muchos casos, los alumnos no vuelven a tener contacto con este tipo de herramientas, ya que no disponen de la posibilidad de adquirirlas.

Mediante el uso de la aplicación CGRA, es posible que los alumnos dispongan de las herramientas en cualquier momento y lugar, pudiendo observar de forma precisa sus detalles y su forma de uso, sin la necesidad de adquirirlas. En la figura 7-b se puede observar a estudiantes de la materia Cirugía General utilizando el prototipo de la aplicación.



a) b)

Fig. 6. a) Comparación entre una herramienta y su representación 3D en escala real. **b)** Seis herramientas del set de instrumental quirúrgico básico en AR.



a) b)

Fig. 7. a) Estudiantes de la asignatura Cirugía General observando un set de herramientas en clase. **b)** Estudiantes de la asignatura Cirugía General utilizando el prototipo de la aplicación.

6. Conclusiones y trabajos futuros

Existen múltiples formas de analizar, parametrizar y digitalizar objetos del mundo real y trasladarlos a un entorno digital. Métodos como el escaneo 3D y la fotogrametría

complementan los campos de modelado por computadora 3D con el objetivo de generar objetos digitales de forma rápida y precisa, aplicado tanto a objetos pequeños como una estatuilla o enormes estructuras como un edificio.

Sin embargo, existen superficies con determinadas características sobre las cuáles dichos métodos pueden resultar contraproducentes, o requerir múltiples adaptaciones y técnicas complementarias para su tratado; y aun así existe la posibilidad de que no logren la precisión necesaria para satisfacer los estándares de una aplicación determinada.

El caso de estudio presentado en este trabajo, la aplicación móvil 3D con realidad aumentada denominada CGRA, permite visualizar modelos 3D en escala real del instrumental quirúrgico, pudiendo variar la orientación y el acercamiento, con el fin de visualizar los detalles de las herramientas. Los objetos fueron construidos a partir de métodos de modelado tradicionales, como lo es el uso de un software de modelado 3D. Los métodos de escaneo 3D y de generación mediante IA no fueron probados, el primero de ellos debido al costo elevado de la adquisición de un equipo de este tipo.

Los modelos y la aplicación fueron evaluados por un conjunto de profesionales y de estudiantes de la Facultad de Ciencias Veterinarias, con el fin de validar su uso y recolectar opiniones y sugerencias que permitan mejorar su calidad.

Actualmente se está desarrollando la segunda versión de la aplicación, en la cual se pretende incorporar nuevos recursos, como por ejemplo videos que muestran detalles del manejo de las herramientas y su utilidad explícita.

Referencias

1. R. Azuma. “A Survey of Augmented Reality”. UNC, 1997.
2. General Surgery Instruments. https://general-surgery-instrument.web.app/.
3. Surgical Instruments. https://play.google.com/store/apps/details?id=com.medical.surgery.instruments.
4. AR Atom Visualizer for ARCore. https://stem.signalgarden.com/.
5. A. Cao, S. Dapoto, P. Thomas, A. Blasco, H. Baschar, J. Terminiello. “Realidad Aumentada Aplicada al Estudio de Instrumental Quirúrgico para Cirugía General Veterinaria”.
6. Modelo de Urotherium antiquus - Museo de Cs. Naturales de La Plata. https://skfb.ly/6zHor.
7. Blender. https://www.blender.org/.
8. FreeCAD. https://www.freecad.org/.
9. AliceVision Photogrammetric computer Vision Framework. https://alicevision.org/.
10. Alpha3D. https://www.alpha3d.io/.
11. D. Slatter. “Tratado de Cirugía en Pequeños Animales”, Ed. 3ra., 2006.
12. Agisoft Metashape. https://www.agisoft.com/.
13. Blender Reference Manual. https://docs.blender.org/manual/en/latest/.
14. Surfaced Studio Academy. 3D Modelling and Animations in Blender. https://academy.surfacedstudio.com/p/3d-modelling-and-animations-in-blender.
15. Catmull-Clark and Loop Subdivision. https://web.cse.ohiostate.edu/~dey.8/course/784/note20.pdf.
16. Unity Real-Time Development Platform. https://unity.com/.
17. Unity’s AR Foundation Framework. https://unity.com/unity/features/arfoundation.
18. AR Core. https://developers.google.com/.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

An integration of Brain-Computer Interfaces with an Alternative Augmentative Communication System

Javier F. Díaz[1], Laura A. Fava[1], Ivana Harari[1] ,
Fernando Martínez[1], Miguel Tellechea[1],

[1]Laboratorio de Investigación en Nuevas Tecnologías Informáticas -LINTI-
Facultad de Informática, Universidad Nacional de La Plata
La Plata, Buenos Aires, Argentina
{jdiaz, lfava, iharari}@info.unlp.edu.ar,
fernando.martinez@live.com.ar, tellecheamiguel@gmail.com

Abstract. People who cannot communicate with others or that have great difficulty in doing so are usually dependent beings and suffer from social isolation. There are multiple systems and tools that enhance the communicative interactions, called Alternative and Augmentative Communication Systems (AACS). In critical situations, where no voluntary movements are available, Brain-Computer Interfaces (BCI) start to take part. BCI's are devices that allow a person to interact with a computer using only their neuronal activity. BCI's have been studied for more than 30 years, but their high error rate and the complexity of the systems have always been challenges when implementing functional real-life solutions. The current work includes a review of the BCI Emotiv Epoc, a wireless, low-end, EGC device that tries to bring medical-lab technology to the users' home, as well as the development of a custom application to connect it to a well-known AACS: ACAT.

Keywords: Brain-Computer Interface, BCI, Alternative and Augmentative Communication, AAC, ACAT, alpha rhythm.

1. Introduction

According to Escandell [1], communication is a complex activity involving various types of entities, representations and processes: the sender intentionally produces a symbolic expression and offers it as an indication of his communicative intention. The receiver, on the other hand, has to decode the linguistic expression and infer the relationship between the indicative clue and the sender's communicative intention in order to recover the representations that the sender wanted to communicate, using heuristic processes that try to find a plausible explanation from the available data.

These data include individual representations, but also representations widely shared by the members of a given culture and/or social group, which largely condition the way of perceiving and reacting to the environment. There are several types of problems that can affect the communicative ability of a person: speech disorders (like dyslalia, dysgnosia, rhinolalia, Dysphemia, tachyphemia, dysphonia or aphonia), oral language disorders (like language delay, aphasia) or written language disorders.

People who cannot communicate with others or that have great difficulty in doing so are usually dependent beings and suffer from social isolation. There are multiple systems and tools that enhance the communicative interactions, called Alternative and

Augmentative Communication Systems (AACS). AACS complement oral language when all by itself is not sufficient for effective communication with the environment, and replace oral language when it is not understandable or absent.

In critical situations, where the person has no voluntary movements or they are severely restricted, Brain Computer Interfaces (BCI) enter the scene. BCIs are devices that acquire and measure brain activity, process and classify these signals to produce a digital output that can be used to perform tasks such as moving a cursor, using a computer or enabling/disabling external actuators (like steering a wheelchair) allowing the person to interact with a computer using only their neuronal activity.

The present article has two main axys: The first one centered on an analysis of different software AACS and the Emotiv Epoc BCI, and the second on the development of a prototype that integrates the BCI with ACAT. Also, it explains the difficulties found and the main solutions developed to achieve this complex process.

2. Motivation

BCI's have been studied for the last 30 years with the main objective of providing assistive technology for people with severe physical functional diversity, but the complexity, slowness and the high error rate have been challenges when implementing real-life solutions. The recent progress in biosensor technology and computing scale and performance have improved the position of BCIs, making them not only viable as assistive technologies but also plausible of being used as everyday artifacts. They have transcended the limits of the lab and experimental technology and are now offered as commercial home-products or even toys.

During the last few years, we interviewed people from different organizations focused on the assistance and integration of people with different physical ailments and functional diversity, and we reached the conclusion that there's a severe lack of information regarding these technologies, their possibilities and the solutions they can provide, specifically about brain-computer interaction. Is difficult to find public or private institutions that develop new hardware, software or that contribute systematized knowledge about existing solutions.

In this context the motivation of the present work is to stimulate the development of these technologies in our near community through our university, the consolidation and systematization of knowledge about existing technologies with a multidisciplinary approach, collaborating with different institutions of our city.

3. Alternative and Augmentative Communication

As we stated earlier, there are several oral language disorders, cognitive or physical restrictions and sensory functional diversity that may affect the communicative ability of a person, and there isn't a generalized solution that works for every person.

Tamarit [2] defines AACS as logopedic instruments for people with different language or communication disorders; their objective is teaching a structured set of non-linguistic codes that would provide the person with the means to communicate

spontaneously and autonomously in combination or not with other vocal or non-vocal codes. It uses different types of expression than traditionally spoken or written language and their purpose is to augment or counteract the difficulties in linguistic capacity of people with functional diversity [3].

The first categorization we can establish is differentiating Augmentative from Alternative Communication. The first one refers to systems designed to enhance a person's speech ability. The other one refers to systems designed to replace speech.

Based on the kind of external support the system needs, there are 2 types of communication possible: Communication with help and communication without help.

Communication without help occurs when the communication act is performed with the person's own body, without assistance of external object or medium. These kinds of systems are more frequently used by people with some kind of sensory functional diversity, like blindness or deafness. The most common communicative systems in this category are sign language, lip-facial reading and the Braille system.

Communication with help occurs when the communication act is performed using some external support. This kind of communication usually uses non-vocal, graphical communication systems, with symbols, and are usually used when the person has a

physical functional diversity (like progressive neuromuscular diseases, cerebral palsy, cranioencephalic trauma) or intellectual functional diversity. Examples of this kind of systems are the Bliss system (a symbolic graphical system) or SPC (spanish acronym for Pictographic System of Communication).

In the present work we will center in Systems that provide Communication with help. Particularly, we will analyze ACAT [4], a free, open-source, suite of applications specially developed by Intel for the physicist Stephen Hawking.

ACAT stands for Assistive Context-Aware Toolkit, is an open source platform developed by researchers at Intel Labs to enable people with motor neuron diseases and other disabilities to easily communicate with others through keyboard simulation, word prediction and speech synthesis. Users can perform a range of tasks such as editing, managing documents, navigating the Web and accessing emails, using a single Keyboard event (by default, F12 key press).

In Fig.1 (a) we can see its two main features: the Talk window (where the user writes what they want the system to read out loud), the Alphabet Scanner (where the user selects either a word from the text predictor or a letter from the alphabet to be added to the text).

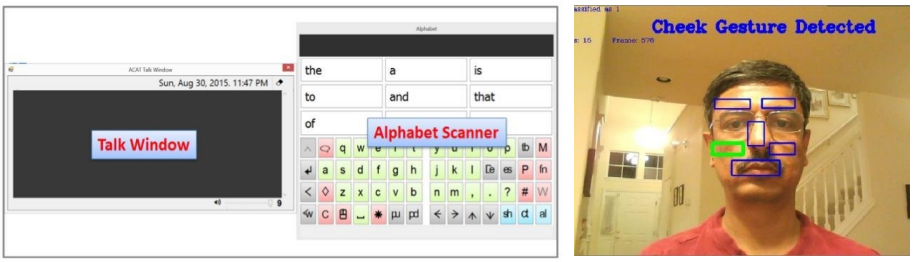


Fig. 1. (a) ACAT application board - (b) ACAT detected gesture región

ACAT also provides ACAT Vision, which acts as a virtual switch using face-tracking and movement detection algorithms on different parts of the face that can be used via an inexpensive webcam (see Fig. 1. (b)). ACAT is open sourced and provides extensive and complex features, so that it can be difficult to configure, setup or extend correctly for non-technical users.

4. Brain-computer Interfaces

A Brain-Computer Interface is a device that allows direct communication between the brain and a computer based on the brain's electrical activity. Nicolas-Alonso and Gomez-Gil [5] defined brain–computer interface (BCI) as a hardware and software communications strategy that empowers humans to interact with their surroundings with no inclusion of peripheral nerves or muscles by utilizing control signals produced from electroencephalographic activity.

By observing and mapping the brain electrical activity to images it is possible to identify patterns, zones in charge of certain cognitive processes. One of the more used and less invasive techniques to observe these is the electroencephalography (EEG) [6]. EEG records voltage fluctuations due to the flow of ionic current during synaptic excitations in the neurons of the brain [7]. They are recorded using electrodes attached to the scalp. Series of the reading of an electrode are called brain signals.

EEG is a non-invasive technique, is not expensive, and has a very good temporal resolution in the order of milliseconds.. It has two main drawbacks: the first is that it has poor spatial resolution as each electrode measures the activity of a big number of neurons; the second is that the signal quality tends to be noisy as it needs to pass through bone and skin and is easily interfered by external electro-magnetic radiation.

Anyway, these characteristics have made EEG the most popular modality among the BCI research community. The EEG brain signals can be categorized according to frequency bands: Delta (from 0.5Hz to 4Hz), Theta (from 4Hz to 8Hz), Alpha (form 8Hz to 13Hz), Beta (from 13 Hz to 30Hz), Gamma (higher than 30Hz). For this work, we are particularly interested in the alpha waves or alpha rhythm. Alpha waves can be measured reliably in the occipital lobes approximately one second after the person closes their eyes(see Fig. 2).

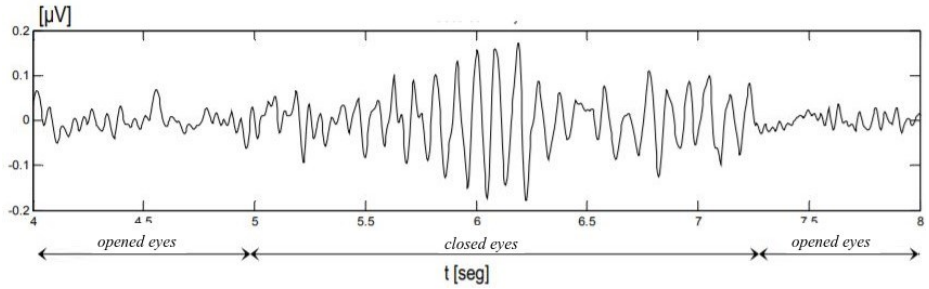


Fig. 2. Alpha rhythm visualization

It was first thought that the rhythm was a representation of the activity of the brain while the visual cortex was in stand-by mode, but more recent studies show that they either inhibit areas of the cortex that are not being used or are related to network communication [8] [9]. It is one of the strongest waves detectable on EEG, they are easy to produce and detect. This makes them frequently used for testing and diagnosing BCI hardware and software.

5. Emotiv Epoc

Emotiv Inc. is an Australian company that develops EEG-based Brain-Computer interfaces. The interface used in the present work is the Emotiv Epoc, recommended for investigation and complex applications (See Fig. 3 (a)).



Fig. 3: (a) Emotiv Epoc / (b) Emotiv Epoc Flex

The EEG has 14 channels using sensors AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4 and 2 references: CMS/DRL references at P3/P4; left/right mastoid process alternative. Sensor's material is saline soaked felt pads.

These should be cheap to produce and easy to replace, but our experience shows that the design is fragile, the pad supports are easy to break. A set of replacement pads cost us half the price of the whole device. The constant use of saline solution tends to corrode the electrical contacts thus making the replacement unavoidable in the mid term. To extend the life of the set we disassemble and thoroughly clean each pad after use, but the work is tedious and not very effective (Eg. by the end of this investigation, brand new electrodes were almost ready for a replacement again).

It is wireless and connects via low energy bluetooth using a custom 2.4GHz USB receiver. Having no cables hanging from one's head is definitely good, but the headset is not really "mobile". The connection is highly affected by distance and we observed the device needs a direct line of view from device to receiver to perform decently.

The EEG signals are acquired by sequential sampling using a single ADC and a sample rate of 128 samples per second (SPS). It could be raised to 256 SPS setting a special configuration in the hardware. The resolution it manages to obtain the data is 14 bits (16 bit ADC, 2 bits instrumental noise floor discarded). It also provides 9 axis motion sensors to detect head movement using an accelerometer and a magnetometer.

The software and drivers run on Windows, Mac, IOS and android (there's no official Linux support, but there are some reverse engineered drivers that support the basic

functions). The suite provides some neat screens to help with the setup of the headset where one can see the quality of the signal of every sensor, the status of the wireless link and battery of the device (See Fig. 4 (a)).

This app works really well and is really helpful, is always needed as part of the setup and lets you avoid re-implementing this kind of utilities in your custom app.

The software suite also provides detection of mental commands: neutral + up to 4 pretrained items per training profile. This is a really interesting feature but after training and using it for several hours, we couldn't find a way to reliably use the commands. Training 2 commands is exhausting. Training more is almost impossible: it looks like the training of a 3rd or 4th command somehow interferes with the detection of the first ones, so training more than 2 is very hard. It also calculates some metrics like excitement, engagement, relaxation, interest, stress and focus.

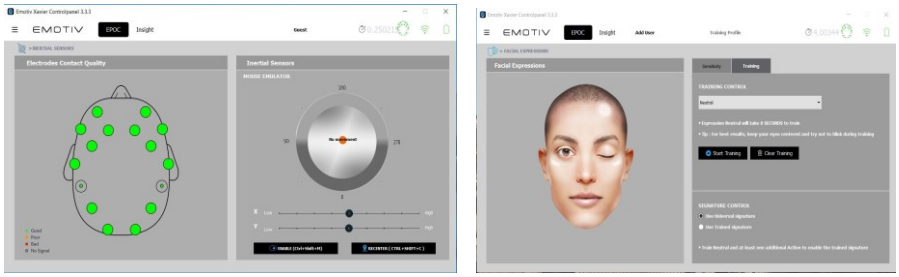


Fig. 4: (a) Signal sensors / (b) Facial expressions detected

It is also able to detect facial expressions like blinking or winking, surprise, frown, smile, clench, laugh and smirk (See Fig. 4 (b)). These detections work better than the mental commands but they are still flickery and our tests were full of false-positives (Eg. a blink benign triggered while the testing subject has his eyes wide open), with a proper noise reduction technique they should be usable to build some interesting interactions with the machine.

Finally, the most interesting feature for this work, although perhaps the more basic one: the pro license (paid) provides access to raw EEG signals, which is what we are going to use to build the integration with ACAT in the next section.

6. Custom Application Development

The objective of the prototype implemented here is twofold. On one hand, be the first practical approach to the Emotiv Epoc and its SDK in our university. On the other, serve as the base for future research and applications development.

So, the work focus is not just functionality, but also tooling. The application will read the raw signals from the Emotiv, process them to generate interaction events and send them to ACAT. We decided we were going to use the detection of alpha rhythm as the source of the events. From a technical standpoint, alpha rhythm is easy to detect with a high degree of confidence. Thanks to this, it is very used as a diagnosis tool and

is frequent to find alpha rhythm detection on BCI software for hardware diagnosis purposes.

6.1 Signal Processing and Digital Filters

As we already mentioned, the signal received from an EEG is noisy. So the first step to use them is to process them to clean them and transform them into something useful. For this we will use a set of digital filters and transformations. Digital filters are mathematical operations that take a series of values (input signal) and modify it in some way and produce another sequence of values (output signal) with the objective of emphasizing or attenuating certain aspects of the input signal.

The algorithm we developed to detect the alpha rhythm and generate the keyboard event has 2 phases: a signal processing phase that consists of a set of composed filters functions (the output of one is the input of the next), and a detection phase, that analyzes the output of the processing phase to determine if the conditions to trigger an event are met using 2 configurable parameters (threshold and sensitivity).

6.2 Implementation

In order to verify the signal processing phase we developed a test mode that you can switch on/off at will. The advantage of the test mode is that it doesn't need the headset connected so we could avoid all the setup and sensor-caring activities.

When the test mode is turned on, the system takes the input signal from an alpha rhythm generator we programmed into the application. The generator creates a synthetic signal which mimics the synchronous activity of the neurons of the occipital lobe of a person opening and closing their eyes for several seconds in an endless loop. The test signal consists of a 10 Hz sinusoidal wave with a tiny amount of noise, which is interrupted by several seconds of white noise. Finally we add some low frequency sinusoidal movement (See Fig. 5).With this in place we could easily develop and test the filter chain and the detection algorithm.

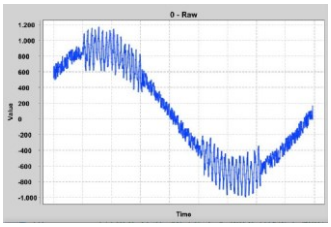


Fig. 5: The original input signal

The signal processing steps are:

- High-pass filter: This step removes all the low frequency noise. You can clearly see the signal has lost the low sinusoidal movement (See Fig. 9 (a)).
- Band-pass filter (8Hz - 12Hz): This step reduces considerably the amplitude of frequencies lower than 8Hz and higher than 12Hz, thus isolating the Alpha band we are interested in (See Fig. 6 (b)).
- Rectifier: This step converts the negative half of the wave to positive, so we now have a positive-only signal forming "10Hz mounds" (See Fig. 6 (c)).

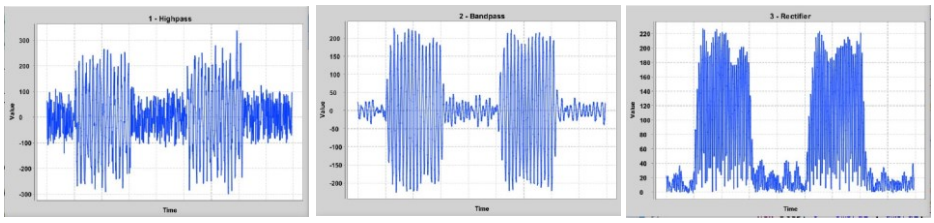


Fig. 6: (a) high-pass filter / (b) band-pass filter / (c) rectifier

- Low-pass filter*:* Here we remove the high-frequency movement and we keep only the profile of the mounds (See Fig. 7 (a)).
- Amplifier: This step elevates every value of the signal to the power of 2, thus making the amplitude of each 10Hz mound bigger, emphasizing the difference with those parts of the signal where we originally had noise (See Fig. 7 (b)).
- Threshold-based square wave conversion: in this last step, the signal is transformed into a square wave of amplitude 1 using a configurable threshold. Those places where the amplitude was greater than the threshold are converted to 1, those lower than the threshold to 0 (See Fig. 7 (c)).

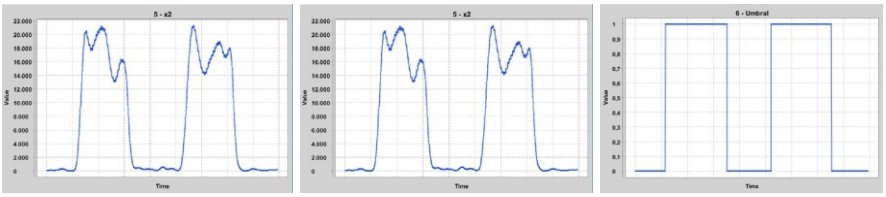


Fig. 7: (a) low-pass filter / (b) amplifier / (c) threshold-based square wave conversión

Finally, we can see the whole filter chain in action processing a real EEG signal taken from the Emotiv Epoc (see Fig. 8).

The subject closed their eyes 2 times in this sample (one on the very beginning of the sampled data, and the other near the end). Notice that the first event has a double peak mound, this kind of artifacts is quite common but the detection phase takes care of cleaning the out.

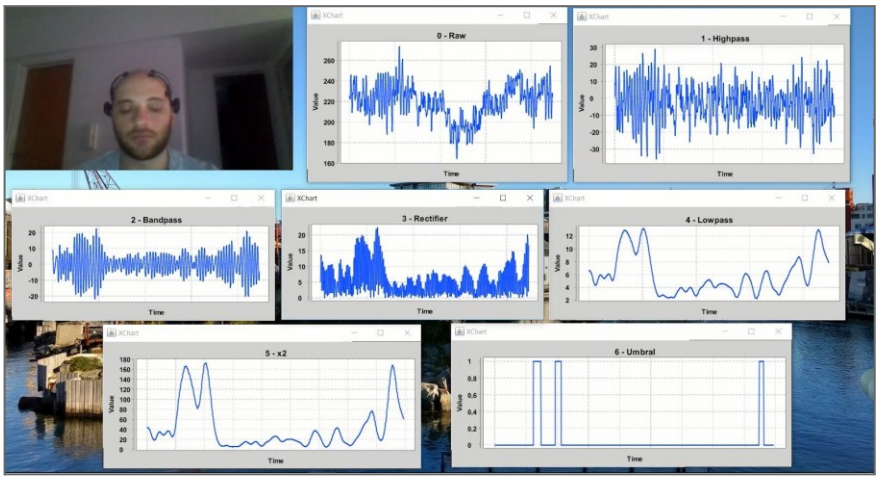


Fig. 8: Filter chain applied to a real EEG signal.

With the signal left by the last step of the signal processing phase the event detection is quite straight forward: counting consecutive 1's. The system then calculates the rate of 1's per second, and triggers a keyboard event and plays a sound if the rate surpasses a given amount.

When we started testing the first implementations of the detection algorithm, we found out that amplitude of the neural activity varies greatly from person to person or even from sessions to sessions of the same person on different days. This introduced the need to add a configuration tool from which we could easily control the detection algorithm parameters and quickly adapt the application responsiveness for the user's situation. The configuration tool developed has 2 parameters: threshold, that is the value of the threshold used in step 6 of the processing phase and sensitivity, that represents the rate of 1's the algorithm must detect in a given second before triggering an event. The values for the threshold can only be obtained by connecting the headset doing 2 or 3 tries and using an average value. We found out that a value around 60% of the max amplitude of the mounds in step 5 works ok most of the time. Regarding sensitivity, usually what is preferable of this type of interfaces is that they were hard to trigger but precise, rather than being full of false-positive non desired "clicks". Thus, having values above 75% is what we found more effective. The 2 parameters have a deep inter-relationship but proved to be sufficiently flexible. For example having 100% sensitivity means that we would need a complete second full of consecutive 1's, which due to the noisy nature of the EEG signals might be hard to obtain, unless you use a very low threshold.

6.3 Integration Testing

ACAT is very powerful, it provides a lot of tools and different mechanisms to control the computer and its programs. All of these tools are available to any device that can trigger a keyboard event (F12 keypress, by default).

The downside is that the configuration is also huge, and knowing how to really use every one of the applications can take a lot of time. Another drawback we found, is that the configuration of the screen scanners delay has a maximum allowed value of 3 seconds and our detection algorithm has a latency between 3 and 5 seconds. This is due to the nature of the alpha rhythm (a person needs to close their eyes for 1 or 2 seconds for it to appear) and the amount of samples we are using to trigger the keyboard event to prevent false positives. This makes us think ACAT is built with a faster and more precise triggering mechanism (for example, Stephen Hawking used it with an infrared sensor in his cheek, which reacts in the order of milliseconds). So the tests we performed resulted in an extremely wearing experience.

Functionally speaking, the integration was seamless, needing a single keypress makes the task really easy. The integration was usable in an ACAT Speller, a training app, that allows up to 5 seconds delay per option in the screen scanner. Also tried it with some board games like tic-tac-toe, and the application gave success results.

It is still a slow reacting control interface. We think the combination of a faster evoked potential (like p300 for example, which takes around 300 milliseconds to appear) and the ACAT intelligent autocompleter (which stores the most used words in a personal dictionary) is viable to have a better user experience and a more responsive interface.

7. Conclusions

This paper presents a custom application development to make an integration of Emotiv Epoc with ACAT tool. This prototype reads the raw signals from the Emotiv, processes them to generate interaction events, and sends them to ACAT. The algorithm was developed to detect the alpha rhythm and generate the keyboard event in 2 phases: a signal processing phase, and a detection phase.

While doing this investigation and development with social implications, we found the complexity of this kind of technology and their integration in order to help people to better communicate using these systems. Applications like ACAT are really complex, not easy to use or configure and have a steep learning curve for non-technical people. But once this curve is surpassed the amount and quality of the tools it provides is excellent.

On the BCI side, we find the Emotiv EPOC not a good fit to use as a every-day home device. The electrodes dry up really quickly. In order to moisten them again, you have to remove the headset and then recheck the connectivity and quality signal of each of them. Signal quality and wireless range is not really an enabler as the user cannot be far away from the computer. Perhapsa wired device could improve the signal quality and stability. work performing the integrations with AACS.

More testing and validation processes where to analyze the use and impact of this technology will be done in future works.

Finally we want to conclude that learning about filters, their implementation and application, and the integration with software such as ACAT was very enriching. We believe that this research is the starting point in our University for the advance of BCI development.

References

1. Escandell Vidal, M. V.: La comunicación. Lengua, cognición y sociedad. Madrid: Akal. pp. 144-145. ISBN: 978-84-460-3958-7. Editorial: AKAL (2014).
2. Tamarit Cuadrado, J.: Sistemas de comunicación, 17-42. Madrid, España. ISBN 84-87699-67-7 (2003).
3. Fernandez-Pacheco, L., Peña Ruiz, B. and Tercero Cotillas, M.: Sistemas aumentativos y alternativos de comunicación. ISBN: 978-84-9077-348-2 Madrid: Editorial Síntesis (2016).
4. ACAT Homepage, https://www.intel.com/content/www/us/en/developer/tools/open/acat/overview.html, last accessed 2023/01/30.
5. Nicolas-Alonso, L.F. and Gomez-Gil, J.: Brain Computer Interfaces: A Review. Sensors, 12, 1211-1279. http://dx.doi.org/10.3390/s120201211 (2012).
6. Zion-Golumbic, Elana: What is EEG?, Hebrew University of Jerusalén. https://www.mada.org.il/brain/articles/faces-e.pdf (2019).
7. Clerc, M., Bougrain, L. and Lotte, F.: Brain-Computer Interfaces 2, Technology and Applications. ISBN : 9781848219632, iSTE & WILEY (2016).
8. Palva S., Palva J.M.: New vistas for a-frequency band oscillations. *Trends Neurosci*. 30 (4): 150–158. doi:10.1016/j.tins.2007.02.001. PMID 17307258. S2CID 9156592 (2007).
9. Rashid, M. Sulaiman, N., Majeed, A. Musa, R. Nasir, A. and Khatun, S.: Current Status, Challenges, and Possible Solutions of EEG-Based Brain-Computer Interface: A Comprehensive Review, Frontiers in Neurorobotics, https://doi.org/10.3389/fnbot.2020.00025 (2020).

Coordinadores:
- Pablo Fillottrani (UNS)
- Marcelo Estayno (UNSAM)
- Alicia Mon (ITBA)

CACiC 2023

CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

XVI Workshop de Innovación en Sistemas de Software

> Short Papers

WISS

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Itinerario: una Apps de Nueva Generación

Adriana Elizabeth Martín[1] ; Susana Beatriz Chavez[2];
Sergio Rafael Flores[3] ; A. Sara Zogbe [4]

Departamento e Instituto de Informática - F.C.E.F. y N. - U.N.S.J.
Complejo Islas Malvinas. Ignacio de la Roza y Meglioli. C.P. 5402.
Rivadavia. San Juan, 0264 4234129

[1] arianamartinsj@gmail.com; [2] susana.chvz@gmail,com;
[3] sergiorflores@gmail.com; [4] aszogbe@gmail.com;

Resumen La computación móvil ha avanzado notablemente en los últimos años, surgiendo a nivel de desarrollo millones de apps de la más variada utilidad, área de aplicación y costos.

Muchas empresas y organizaciones ofrecen soluciones para sus usuarios que son potenciadas en nuevas versiones. La pandemia aceleró el uso de los smartphones, que pasaron a ser de gran utilidad, lo que potenció el uso de la red y los dispositivos, permitiendo el surgimiento de nuevas aplicaciones a partir de tecnologías emergentes como: La redes 5G y 6G, las superapps, aplicaciones móviles basadas en Al, la computación inmersiva, entre otras.

Las nuevas aplicaciones necesariamente deben conectarse al cloud, de donde surgen actualizaciones, atención de requerimientos u otras funcionalidades. Serverless Computing es una arquitectura para Cloud Computing que presenta ventajas considerables para integrarse con la computación móvil, por lo que es la arquitectura utilizada.

Palabras claves: Mobile Computing. SupperApps, Inteligencia Artificial. Computación Inmesiva. Serverless

1. Introducción

En los últimos años, la tecnología móvil ha avanzado rápidamente, y algunas de las tendencias más notables incluyen la adopción de redes móviles más rápidas, el surgimiento de las superapps, la computación inmersiva y las aplicaciones móviles basadas en inteligencia artificial (IA). La quinta generación de redes móviles (5G), adoptada en muchos países, permite velocidades de descarga mucho más rápidas y menor latencia, esto significa que los dispositivos pueden conectarse y comunicarse entre sí en tiempo real.

Superapps

Las superapps ofrecen una amplia gama de servicios y funciones en una sola plataforma, lo que las convierte en una opción conveniente para los usuarios que buscan simplificar sus vidas digitales. En lugar de tener varias aplicaciones para realizar diferentes tareas, como pedir comida, hacer una reserva de hotel o comprar productos

en línea, reservar taxis hasta comprar entradas de cine y enviar dinero a amigos y familiares, una superapp ofrece todo esto y más en una sola aplicación. Incluso se pueden realizar pagos sin tener que salir de la aplicación. Esto convierte a las superapps en una opción conveniente para los usuarios que buscan simplificar sus vidas digitales.

Computación inmersiva

Este concepto se refiere a una experiencia informática en la que el usuario se sumerge en un entorno generado computacionalmente a través de interfaces de usuario avanzadas y dispositivos de entrada/salida como pantallas 3D, gafas de realidad virtual, guantes con sensores, y otros dispositivos de interacción. La idea detrás de la computación inmersiva es crear un entorno informático que permita a los usuarios interactuar de manera natural y fluida con el mundo virtual, sintiéndose como si estuvieran dentro de él. Esto se logra mediante el uso de tecnologías avanzadas, como la realidad virtual (VR), la realidad aumentada (AR) y la realidad mixta (MR). El entorno de la tecnología inmersiva presenta diferentes características como fruto de un progreso del continuo avance del hardware , lo cual permite que dichas tecnologías se vuelvan accesibles, permitiendo la unión de diferentes tecnologías.

Servicio Serverless

Serverless es una arquitectura de computación en la nube que está ganando popularidad en el mundo de las aplicaciones móviles de nueva generación. En lugar de tener que gestionar y mantener infraestructuras complejas y costosas, el desarrollo de aplicaciones serverless permite que los desarrolladores se centren en escribir código de aplicación sin preocuparse por la infraestructura subyacente. Esto se logra mediante la externalización de la gestión de servidores a los proveedores de servicios en la nube, como Amazon Web Services (AWS). AWS Amplify es un conjunto de herramientas y servicios que permite a los desarrolladores crear y escalar aplicaciones móviles sin tener que preocuparse por la infraestructura subyacente. Amplify proporciona una plataforma completa para la construcción de aplicaciones móviles que incluye funcionalidades como la autenticación de usuarios, el almacenamiento de datos, el análisis y la integración con otras herramientas y servicios de AWS. Además, Amplify es compatible con una amplia gama de tecnologías y marcos de trabajo, lo que lo hace flexible y escalable. Actualmente, es compatible con iOS, Android y JavaScript (web y React Native).

2. Motivación

Durante los últimos años, el turismo se ha convertido en una de las actividades con mayor crecimiento a nivel mundial. Pero no sólo trae ventajas desde el punto de vista económico, sino que además fomenta la mejora en otros sectores tales como el cultural, el social y político, el ambiental, entre muchos otros. En este trabajo se está diseñando y desarrollando un prototipo de superApp turística en el Cloud que permite la incorporación de contenido multimedia audiovisual llamada “Itinerario”.

Dicha aplicación incorpora un mapa interactivo con geolocalización del recorrido habitual que se realiza asociado a la actividad turística. Desde allí se accede a otras apps como la red Tulum (aplicación de transporte de colectivo de la provincia de San Juan) para tener una alternativa de llegar al lugar, hace reservas de restaurant y/o hospedaje. Permite el pago de entradas , acceso a realidad aumentada. Tiene acceso al clima, y permite acceder a un traductor ,como muestra la figura 1

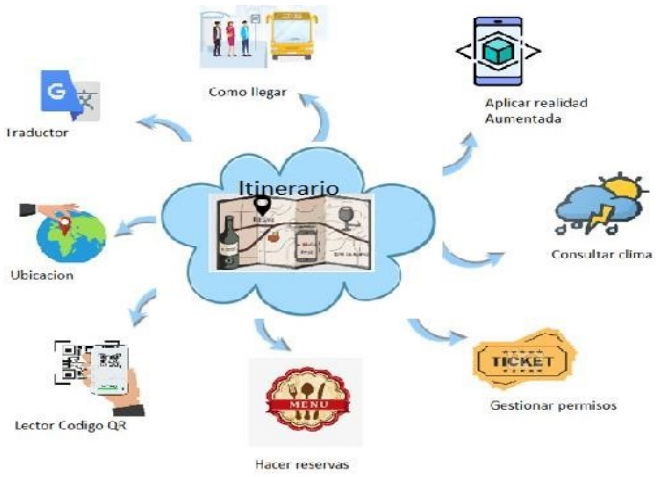


Figura1 Diagrama App

En la actualidad existen algunas aplicaciones para dispositivos móviles que reúnen algunas características de las explicadas en el presente trabajo, como por ejemplo BA Turismo, GIDA o Audioguiame. Sin embargo, dichas aplicaciones no ofrecen la versatilidad de adaptarse a un contenido científico específico ni tampoco poseen libre acceso al código fuente, lo que permitiría adaptar a los requerimientos de cada institución. Otra desventaja que poseen es, en muchos casos, un elevado costo de acceso para el usuario.

La plataforma de desarrollo utilizada es Flutter ya que posee un conjunto de herramientas que permite crear aplicaciones móviles, web y de escritorio, compiladas de forma nativa a partir de una única base de código. Además, Flutter y AWS Amplify del Cloud permite a través de Amplify Flutter, la integración con Flutter. Dando lugar a un espacio integrado de trabajo con todo el poder de AWS. La arquitectura serverless es una forma innovadora y rentable de construir aplicaciones móviles de nueva generación, y AWS Amplify es una solución completa que ofrece a los desarrolladores una manera fácil de construir, escalar y administrar aplicaciones móviles.

En la figura 2 se muestra la página de configuración de la Superapp “Itinerario”.

3. Contexto

El presente trabajo se enmarca dentro del proyecto de investigación: “Soporte Serverless para aplicaciones móviles de nueva generación” , que ha sido aprobado en

la última convocatoria por CICITCA , UNSJ. y está en desarrollo para el período 2023-2024. Las unidades ejecutoras para dicho proyecto son el Departamento e Instituto de Informática de la FCEFyN de la UNSJ.



Figura 2 App Itinerario

4. Aportes de trabajo

Como ya se mencionó, en los últimos años, el turismo se ha convertido en una de las actividades con mayor crecimiento a nivel mundial. Toda esta situación implicó un desafío para el sector turístico, ya que ha tenido que reinventarse teniendo en cuenta novedosas innovaciones y requerimientos en base a las exigencias del nuevo mundo. El auge de Internet y la tecnología, como así también la llegada de los dispositivos móviles y la posibilidad de descarga de aplicaciones en ellos, ofrecieron a los turistas una amplia gama de beneficios. Este trabajo está pensado como aporte al turismo provincial, con la intención que sea utilizado ya sea como recorrido de algún museo, o bien la ruta del vino o ruta del Olivo, o cualquier actividad que pudiera surgir en el futuro. Posee la ventaja de poder tener integrado en una sola app una variedad de funcionalidad no solamente propia de la aplicación, sino integrar aplicaciones de terceros.

5. Formación de recursos humanos

El equipo de trabajo de esta línea de investigación está compuesto de seis investigadores que figuran en este trabajo de las universidades Nacional de San Juan y dos alumnos de grado. Además, el proyecto marco donde se está desarrollando esta

propuesta ha establecido vínculos con investigadores de la Universidad Nacional de San Luis, la Universidad Champagnat y la Universidad Nacional de Salta y dos alumnos de grado.

Se está desarrollando una tesis doctoral sobre paralelismo híbrido y Big Data, una tesis de maestría en áreas afines y dos tesinas de grado en el área de Serverless computing, Concurrencia y Computación distribuida. Además, se espera aumentar el número de publicaciones. Por otro lado, también se prevé la divulgación de varios temas investigados por medio de cursos de postgrado y actualización o publicaciones de divulgación y asesoramiento a empresas y otras instituciones públicas y privadas.

6. Investigaciones futuras

Por el momento la aplicación se está desarrollando bajo Android, pero se prevé a futuro implementar este componente en IOS.

Se prevé en el futuro poder tener acceso a otras app que resulten de interés para el prototipo de manera dinámica.

Momentáneamente se está trabajando con conectividad. Se prevee a futuro implementar la app para que funcione sin conexión a internet y que cuando recupere la conexión vuelva a sincronizar.

Bibliografía Consultada

1. Jonas E, Schleier-Smith … (2019) Cloud Programming Simplified: A Berkeley View on Serverless Computing. http://arxiv.org/abs/1902.03383. Jan 2021
2. S. Eismann et al., 'Serverless Applications: Why, When, and How?' (2021), IEEE Software, vol. 38, no. 1, pp. 32–39, Jan. 2021, doi: 10.1109/MS.2020.3023302. problems. In Research Advances in Cloud Computing (pp. 1-20). Springer, Singapore.
3. Nastic, S., Rausch, T., Scekic… (2017). A serverless real-time data analytics platform for edge computing. IEEE Internet Computing, 21(4), 64-71. [4] Mohanty, S. K., Premsankar, G., & Di Francesco, M. (2018). An Evaluation of Open Source Serverless Computing Frameworks. In CloudCom (pp. 115120).
4. Gottlieb, N. (2016). State of the Serverless Community Survey Results. https: //serverless.com/blog/state-of-serverless-community/.
5. Jonas, E., Pu, Q., Venkataraman, S., Stoica, I., & Recht, B. (2017). Occupy the cloud: Distributed computing for the 99%. In Proceedings of the 2017 Symposium on Cloud Computing (pp. 445-451). ACM.
6. Fromm, K. (2012). https://readwrite.com/2012/10/15/why-the-future-ofsoftware-and-apps-is-serverless/
7. Desarrollo de pila completa: AWS Amplify. https://aws.amazon.com/es/amplify/

Coordinadores:
- Fabiana Piccoli (UNSL)
- Laura De Giusti (UNLP)
- Carlos García Garino (UNCU)

CACiC 2023

CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

XXIV Workshop de Procesamiento Distribuido y Paralelo

> Full Papers

WPDP

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Evaluación de la inicialización y el arranque en frio de los lenguajes de programación en una plataforma serverless. Amazon Web Services como caso de estudio

Matías Rodríguez, Nelson Rodríguez, María Murazzo,

Departamento de Informática, Facultad de Ciencias Exactas Físicas y Naturales, Universidad Nacional de San Juan, San Juan, Argentina

matiasdaroni@gmail.com, nelson@iinfo.unsj.edu.ar, maritemurazzo@gmail.com

Resumen. Serverless Computing es una arquitectura en el Cloud, alternativa al modelo tradicional. Ofrece numerosas ventajas sobre el modelo monolítico, como aportar agilidad, innovación, escalado automático, flexibilidad en el desarrollo y una mejor evaluación y control de los costos. Surgió como una evolución de microservicios corriendo en contenedores e implementando funciones, por lo cual a veces se lo denomina función como servicio.

En el presente trabajo se evalúa el comportamiento de distintos lenguajes de programación en la inicialización y el arranque en frío, bajo un enfoque serverless en la plataforma Amazon Web Services. Debido a que la comparación de los lenguajes se puede llevar a cabo en diferentes aspectos o tomando diferentes tipos de métricas, se consideró oportuno realizar las pruebas mediante operaciones CRUD. Esto además permite analizar el comportamiento de los mismos al inicializar la base de datos. Surgen por lo tanto una serie de interrogantes como: ¿La inicialización de los lenguajes de programación son todos iguales?, ¿La inicialización de la base de datos con los lenguajes de programación es independiente del lenguaje usado?, ¿El cold start es equivalente en los distintos lenguajes? ¿El impacto del mismo es relevante cuando la cantidad de requerimientos es elevada? ¿Cuál es la combinación de lenguajes de programación más conveniente para operaciones CRUD?.

Mediante las pruebas realizadas se pudieron evaluar el comportamiento de los lenguajes de programación y contestar algunos de los interrogantes arriba mencionados.

Keywords: Serverless Computing, FaaS, Programming Language, Cloud Computing

1. Introducción

El término Cloud Computing entró en uso popular en 2008, aunque la práctica de proporcionar acceso remoto a las funciones informáticas a través de redes se remonta a los sistemas de tiempo compartido de mainframe de los años 1960 y 1970. La Virtualización creada por IBM, permitió que múltiples sistemas virtuales se ejecuten sobre un solo sistema físico [1]. Luego surge Internet de forma global y su expansión comenzó cuando en 2002, Amazon introdujo sus servicios minoristas basados en la web. Fue la primera gran empresa en pensar en usar solo el 10% de su capacidad (que era común en ese momento) como un problema a resolver [2] [3].

Siguiendo la evolución observada en la historia de la contenerización, los servicios en la nube se han adaptado para ofrecer contenedores de mejor ajuste que requieren

menos tiempo para cargar (arranque) y proporcionar mayor automatización en el manejo (orquestación) de contenedores en nombre del cliente [4]. La Computación Serverless promete lograr la automatización completa en la gestión de contenedores.

El modelo de computación serverless es impulsado por eventos en el que los recursos informáticos se proporcionan como servicios escalables. En el modelo tradicional se cobra un costo fijo y recurrente por los recursos informáticos del servidor, independientemente de la cantidad de trabajo realizado por el servidor. Sin embargo, la implementación Serverless ha superado esta deficiencia, ya que se paga solo por el uso del servicio y no se cobra por el tiempo de inactividad.

En este paradigma emergente, las aplicaciones de software se descomponen en múltiples funciones independientes sin estado [5] [6]. Las funciones solo se ejecutan en respuesta a acciones desencadenantes (como interacciones de usuario, eventos de mensajería o cambios en la base de datos), y se pueden escalar de forma independiente y pueden ser efímeras (pueden durar una invocación) y están completamente administrados por el proveedor de Cloud.

Los principales proveedores de nube han propuesto diferentes plataformas informáticas sin servidor como AWS Lambda, Microsoft Azure Functions, Google Functions, IBM Cloud Functions, Cloudflare Worker, Alibaba Function Compute. Dichas plataformas facilitan y permiten que los desarrolladores se centren más en la lógica de negocios, sin la sobrecarga de escalar y aprovisionar la infraestructura [7].

Castro et al [8], ofrecen una definición basada en las características: “La informática serverless se puede definir por su nombre, que es pensar (o preocuparse) menos por los servidores. Los desarrolladores no necesitan preocuparse por los detalles de bajo nivel de administración y escalado de servidores, y solo pagan cuando procesan solicitudes o eventos”. Luego la define como: “La informática serverless es una plataforma que oculta el uso del servidor a los desarrolladores y ejecuta código que escala bajo demanda automáticamente y facturado solo por el tiempo que se ejecuta el código”.

En la mayoría de los casos, se pueden escribir funciones en el lenguaje que el programador considere más adecuado (Node.js, Python, Go, Java y más) y utilizar herramientas de contenedor y serverless, como AWS SAM o la CLI de Docker, para compilar, probar e implementar las funciones [9].

Por otro lado, especialistas de Expert Market Research pronostican que el mercado global de computación serverless crecerá en el período de pronóstico de 2022-2027 a una tasa compuesta anual del 22.2% [10].

Un modelo basado en funciones es adecuado para ráfagas, uso de CPU intensivo, cargas de trabajo granulares. Actualmente, los casos de uso de FaaS varían ampliamente, incluido el procesamiento de datos, el procesamiento de flujo, la computación de borde (IoT) y la computación científica [3].

Serverless cubre una amplia gama de tecnologías, que se pueden agrupar en dos categorías: Backend-as-a-Service (BaaS) y Functions-as-a-Service (FaaS).

Backend-as-a-Service permite reemplazar los componentes del lado del servidor con servicios listos para usar.Algunos ejemplos son los sistemas de autenticación remota, la administración de bases de datos, el almacenamiento en el cloud.

Existen diversos desafíos, oportunidades y problemas a resolver, entre ellos la experiencia del desarrollador [8], Interoperabilidad, testing, composición de funciones,

seguridad, administración del ciclo de vida, administración de requerimientos no funcionales, performance, optimización del overhead, ingeniería para costo-performance, entre otros [7].

Idealmente, sería deseable tener una sobrecarga mínima al invocar las funciones. Sin embargo, cuando la plataforma necesita activar la primera instancia, los recursos subyacentes aún necesitan tiempo para la inicialización. Este arranque también ocurre cuando el escalador automático aprovisiona instancias adicionales para controlar el tráfico. El tiempo de inicialización es inevitable para cada instancia de función e introduce un retraso hasta que la instancia pueda responder a las solicitudes. Este problema es muy común en plataformas serverless y se conoce como el problema de arranque en frío. Aunque la oferta de FaaS generalmente sufre de arranques en frío, la sobrecarga en la que incurre cada plataforma varía según la implementación subyacente de las funciones [11]

2. Trabajos relacionados

En el desarrollo de software basado en Cloud, los recursos alquilados siempre se pueden mantener activados para servir a la ejecución de la aplicación, que no tiene problemas de arranque en frío para las ejecuciones de aplicaciones. Además, los desarrolladores pueden seleccionar de forma flexible la capacidad de los recursos y configurar el tiempo de servicio de los recursos.

A diferencia, en desarrollo de software basado en serverless, el proveedor es responsable de la administración del entorno de tiempo de ejecución. La ventaja de este tipo de gestión de recursos unificados es responder a cualquier carga de trabajo en ráfaga. Estos entornos de tiempo de ejecución se activan cuando se activan las aplicaciones. Cuando los entornos requeridos no están activos, las aplicaciones pueden enfrentar el problema del arranque en frío, lo que introduce un largo tiempo de preparación. Ha habido muchos esfuerzos para aliviar este problema [12,13, 14, 15, 16, 17]. Sin embargo, ninguno de estos trabajos analiza el comportamiento de los lenguajes de programación en esta problemática.

Un trabajo publicado por [18] es muy claro en la explicación del problema, afirmando lo siguiente: Serverless Computing es el último modelo de computación en la nube, En este modelo la plataforma serverless ofrece elasticidad instantánea por solicitud. Tal elasticidad generalmente se produce a costa del problema de los "arranques en frío". Este fenómeno está asociado con un retraso que se produce debido a la provisión de un contenedor en tiempo de ejecución para ejecutar las funciones. Poco después de que Amazon introdujera este modelo informático con la plataforma AWS Lambda en 2014, Varias plataformas comerciales y de código abierto comenzaron a adoptar y ofrecer esta tecnología. Cada plataforma tiene su propia solución para hacer frente a los arranques en frío. La evaluación del rendimiento de cada plataforma bajo la carga y los factores que influyen en el problema del arranque en frío ha recibido mucha atención en los últimos años.

3. Metodología

La problemática abordada es la evaluación del comportamiento de los lenguajes de programación para la inicialización y el arranque en frío, en una plataforma serverless, tomando como caso de estudio a Amazon Web Services.

Para llevar a cabo los estudios de laboratorio pertinentes sobre diversas variables en la ejecución y puesta en marcha de Funciones Serverless (FaaS), se realizarán las siguientes actividades:

- Análisis y estudio de la plataforma o servicio AWS Lambda.
- Estudio y funcionamiento detallado de las bases de datos disponibles en AWS.
- Determinación de los lenguajes a comparar haciendo uso de funciones.
- Determinación de los escenarios a utilizar que permitan contrastar los valores resultantes.
- Estudio comparativo de los diferentes parámetros y valores obtenidos.

Para el desarrollo de la investigación se procedió como se muestra en la figura 1, la cual consiste en un esquema que sintetiza la labor realizada.

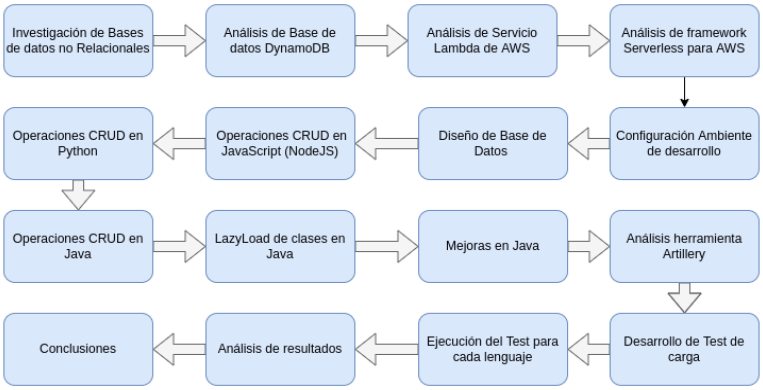


Figura 1 – Esquema del procedimiento del proceso de investigación

Para la investigación se tomó la decisión de desarrollar una API de formularios, utilizando los servicios de AWS que permiten un desarrollo serverless. Se consideró a una API que permita el manejo de distintos formularios dinámicos y que, a su vez, conserve la información pertinente a estos junto con sus respuestas. Lo anterior se pensó adecuado para reproducir un problema del mundo real en el que se necesite de recursos de software escalables cuyos objetivos sean los de soportar grandes volúmenes de usuarios sin afectar el rendimiento del sistema.

Se dispuso realizar una API de formularios, para la comparativa de los distintos lenguajes de programación en un entorno serverless.

La función de esta API es la de crear formularios dinámicos y almacenar toda la información requerida por estos, incluyendo las respuestas de los usuarios. Para realizar la evaluación de los lenguajes bajo este contexto, solamente se necesitó de un formulario. Para la evaluación se implementó un extracto del formulario utilizado para el Censo Nacional de Población, Hogares y Viviendas 2022, dado que este tiene muchos campos y datos de variado tipos.

4. Base de datos y ambiente de desarrollo

Con la finalidad de almacenar los datos necesarios para la representación de formularios y sus respuestas, se optó por utilizar el servicio DynamoDB de AWS ya que permite realizar la investigación bajo el enfoque serverless debido a que este servicio es totalmente administrado y evita las cargas administrativas que supone tener que utilizar y escalar bases de datos. También este servicio dispone de la potencia necesaria para crear tablas que permiten almacenar y recuperar miles y hasta millones de datos y solicitudes.

En esta investigación se propuso imitar de la mejor forma al tráfico de un aplicativo real y de uso constante, por lo que este servicio resulta fundamental para poder replicar el contexto deseado en el cual se evaluará a los lenguajes de programación elegidos.

Para el desarrollo de las APIs de formularios en los distintos lenguajes sobre la plataforma de AWS, se hizo uso de varias herramientas que permiten y facilitan este desarrollo. Las mismas son:

- Github
- Serverless Framework
- AWS CLI
- NodeJS y NPM
- Python y PIP
- Maven
- Java SDK

Algunas de las mencionadas anteriormente son dependencias necesarias para el desarrollo de las APIs dependiendo cada lenguaje de programación. Por ejemplo para JavaScript con NodeJS se necesita instalar NodeJS y NPM, para Python se requiere de Python y PIP, por último para desarrollar con Java se necesita Maven y el SDK de Java.

El desarrollo de este trabajo fue realizado en una notebook personal bajo la distribución Ubuntu 20.04 del sistema operativo Linux.

5. Resultados obtenidos

Para realizar el análisis comparativo entre los lenguajes de programación seleccionados, con el caso de estudio elegido, se realizaron varias experiencias a través de variaciones de test de inicialización. Cada uno de estas variantes está orientado a un objetivo específico y fueron desarrollados y ejecutados mediante la herramienta open source Artillery [19]. De esta manera se evaluaron diversos aspectos, tales como: tiempos de inicialización para las primeras solicitudes a las funciones lambdas, tiempos de ejecución en frío, tiempos de ejecución en caliente y número de solicitudes

El test de inicialización, con todas las variantes mencionadas, permitió el análisis de la performance de los lenguajes bajo distintas situaciones.

Para llevar a cabo el test de inicialización se optó por ejecutarlos con 10 GB de memoria. Este es el tamaño máximo de memoria ofrecido por AWS en el servicio Lambda, lo cual posibilita utilizar la mayor potencia de CPU permitida por este.

5.1 Test de iniciación

El test de inicialización tiene por objetivo probar las operaciones CRUD de tal forma que siempre se produzca la inicialización del contexto de ejecución de cada función lambda. Cuando esto ocurre AWS nos brinda en sus métricas una variable de tiempo de inicialización que indica cuánto le tomó a la plataforma preparar la infraestructura y dependencias necesarias para permitir la ejecución del código dentro de la lambda que fue escrito por el consumidor del servicio.

Se realizaron cinco ejecuciones en frío para cada lenguaje y operación CRUD, esto se hizo mediante la ejecución del flujo del test mencionado en el capítulo anterior. Es importante remarcar que estas ejecuciones se realizaron cada una hora para asegurar que AWS dio de baja la infraestructura FaaS retornando cada función a un estado frío. Cada vez que se hizo una ejecución en frío, también se realizó una segunda ejecución de forma inmediata para tomar el tiempo de ejecución en caliente, lo que permitió realizar una comparativa entre los tiempos de ejecución en frío y caliente.

5.2 Tiempos de inicialización

En la tabla 1, se presentan los promedios obtenidos de los tiempos de inicialización correspondientes a las operaciones CRUD por cada lenguaje de programación.

Tabla 1. Promedio de tiempos de inicialización por cada operación CRUD y lenguaje de programación

Lenguaje	Create	Read	Update	Delete
JavaScript (NodeJS)	418,232	431,04	413,586	442,15
Python	384,902	400,43	361,624	411,2
Java	1623,528	1630,52	1526,12	1577,302

Teniendo en cuenta todas las operaciones CRUD, Se puede ver una clara diferencia, donde la operación read de Java es la que mayor tiempo de inicialización registra con 1630.52 ms. En cuanto a NodeJS y Python la operación más deficiente es delete con tiempos de 442.15 ms. y 411.2 ms. respectivamente. Estas diferencias pueden apreciarse en la figura 2 aunque no son significativas. Siguiendo el mismo razonamiento la operación update es la de mejor rendimiento a la hora de inicializar el contexto de ejecución para una lambda, cualquiera sea el lenguaje.

Como se mencionó anteriormente, también se obtuvieron resultados en cuanto a los tiempos de ejecución en frío y de una segunda ejecución en caliente. Las tablas 2 y 3 y 4 muestran las diferencias entre los tiempos de ejecución en frío y en caliente para cada una de las operaciones CRUD y por cada lenguaje. Las celdas sombreadas corresponden a la operación CRUD que mayor mejora logra. Se puede apreciar que en Java es donde las diferencias son mayores. Por ejemplo, con delete, la ejecución pasa de tardar 581.494 ms. en frío a 84.654 ms. en caliente. Es decir, Java es el lenguaje con mayor discrepancia entre esos tiempos de ejecución, obteniendo una mejora en

promedio del 84%. Dicho de otro modo, Java es el que reduce los tiempos de forma más abrupta. También NodeJS y Python mejoran al ejecutarse en caliente, encontrando las mayores ganancias en Read y Delete, respectivamente, de aproximadamente un 48% en promedio.

Tabla 2. Tiempo de ejecución en NodeJS, por cada operación CRUD

Estado de tiempo de Ejecución	Create	Read	Update	Delete
Frio	76.698	74.592	84.04	77.768
Caliente	51.238	39.248	45.536	41.066
Diferencia	25.46	35.294	38.504	36.702
Porcentaje de Mejora	33.195112	47.347803	45.816278	47.194218

A continuación, se muestra la tabla 3, con los valores de ejecución en frío y en caliente de Python.

Tabla 3. Tiempo de ejecución en Python, por cada operación CRUD

Estado de tiempo de Ejecución	Create	Read	Update	Delete
Frio	60.734	49.752	65.738	43.97
Caliente	32.582	27.388	38.472	22.566
Diferencia	28.152	22.364	27.266	21.404
Porcentaje de Mejora	46.352948	44.950956	41.476771	48.6786445

Y por último se muestran los resultados obtenidos luego de ejecutar las pruebas en el lenguaje de programación Java.

Tabla 4. Tiempo de ejecución en Java, por cada operación CRUD

Estado de tiempo de Ejecución	Create	Read	Update	Delete
Frio	701.192	664.628	644.328	581.494
Caliente	116.638	98.102	116.89	84.654
Diferencia	584.554	566.526	527.438	496.84
Porcentaje de Mejora	83.365754	85.239562	81.858618	85.441982

Por último se muestra en la figura 2, un resumen de los resultados de todas las ejecuciones realizadas, aunque algunas de las diferencias no son significativas.

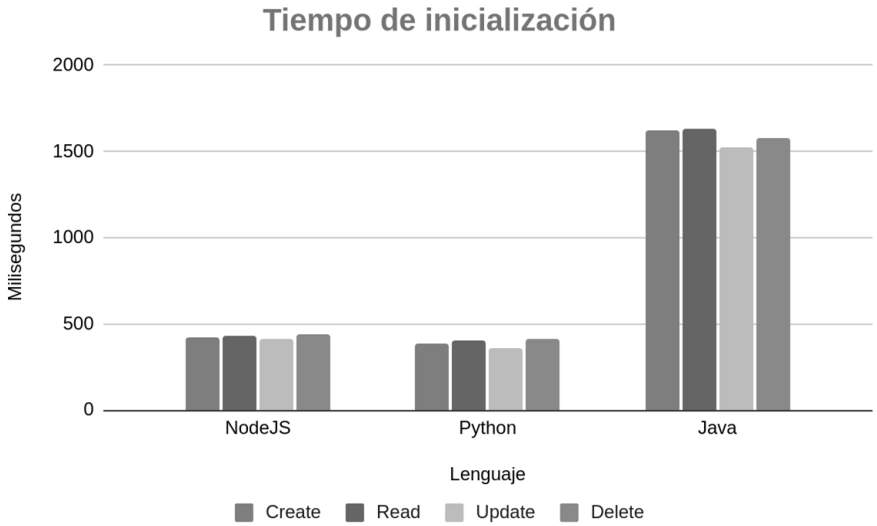


Figura 2. Comparativa de los tiempos de inicialización de operaciones CRUD de los diferentes lenguajes de programación.

6. Conclusiones y Futuros trabajos

En este trabajo se desarrolló una API Rest con el objetivo de realizar operaciones CRUD para la carga de un formulario y sus respuestas. Esta API fue desarrollada en tres lenguajes diferentes: Javascript con NodeJS, Python y Java. Con estas operaciones se investigó, en general, la variación de la performance, la latencia de los tiempos de inicialización, la escalabilidad y las implicaciones del tamaño de memoria configurado en las funciones. Se investigó cómo afecta la selección de los lenguajes de programación a la hora de realizar operaciones CRUD de una API Rest con una serie de test automáticos.

El lenguaje de programación con mejores tiempos a la hora de levantar el contexto de ejecución de una función lambda es Python, obteniendo los mejores tiempos para cualquiera de las operaciones CRUD. De igual forma, las diferencias entre Python y NodeJS son realmente insignificantes. La elección entre cualquiera de estos dos lenguajes, teniendo en cuenta la inicialización de las lambdas, no será muy conflictiva. El lenguaje más deficiente fue Java, teniendo tiempos de inicialización muy grandes con respecto a los demás. Java a diferencia de NodeJS y Python debe levantar una máquina virtual y cargar el código de la aplicación en memoria, lo cual produce que demore su primera ejecución. La elección de Java para una API Rest serverless con Lambda puede ocasionar tiempos de respuesta lentos en caso de usarse en un sistema de poco o mediano tráfico. Para futuros trabajos se puede hacer hincapié en implementar optimizaciones para acotar los tiempos de inicialización de las funciones lambda, desarrolladas con Java.

De acuerdo al test de inicialización también se pudo concluir en que Java, a pesar de ser el lenguaje que peor performance tiene en cuanto a tiempos de inicialización, es el que tiene un mayor rango de mejora de una ejecución en estado frío a una en estado

caliente. La elección de este lenguaje puede estar justificada en un sistema de tráfico moderado-alto donde sea muy poco probable tener funciones lambdas en estado frío. Los rangos de mejora de NodeJS y Python son grandes porcentualmente pero poco significativos en un caso real. Debido a que los tiempos de ejecución bajan un porcentaje grande de estado frío a caliente, se debe tener en cuenta que mientras menos tarden las funciones lambda, menos tiempo será facturado por AWS. Por ende el lenguaje con menor tiempo de ejecución será el más barato. Un caso interesante para trabajos futuros podría ser investigar acerca de las distintas alternativas para mantener las lambdas en estado caliente, siendo una de ellas ofrecida por AWS pero sumamente costosa.

Además de las propuestas a futuro, indicadas en los párrafos anteriores, sería conveniente evaluar el comportamiento de otros lenguajes de programación como Go, C#, Ruby y PowerShell.

Actualmente se están desarrollando evaluaciones de los tamaños de memoria, la variación de la performance de los lenguajes de programación y la escalabilidad. Resultados preliminares de test de memoria han determinado que los tiempos de inicialización de las lambdas implementadas en NodeJS y Python no se ven afectados por el aprovisionamiento de memoria.

Referencias

1. Cloud computing: A complete guide. IBM. https://www.ibm.com/cloud/learn/cloud-computing-gbl
2. M. Armbrust, et al., Above the clouds: A Berkeley view of cloud computing, In: Tech. Rep. No. UCB/EECS-2009-28, 2009.
3. Keith D. Foote: A Brief History of Cloud Computing. https://www.dataversity.net/brief-history-cloud-computing/ (Dec. 2021)
4. E. van Eyk, L. Toader, S. Talluri, L. Versluis, A. Uță and A. Iosup, : Serverless is More: From PaaS to Present Cloud Computing. In: IEEE Internet Computing, vol. 22, no. 5, pp. 8-17, Sep./Oct. 2018, doi: 10.1109/MIC.2018.053681358.
5. Adzic, G., Chatley, R.: Serverless computing: economic and architectural impact. In Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering (pp. 884-889). ACM.(2917)
6. AWS Lambda: aws.amazon.com/es/lambda/
7. Bermbach D., Karakaya A., Buchholz S.: Using Application Knowledge to Reduce Cold Starts in FaaS Services. In: SAC ’20, March 30-April 3, 2020, Brno, Czech Republic (2020).
8. Castro p., Ishakian v., Muthusamy v., Slominski a.: The rise of serverless computing. In: Communications of the ACM | Dec. 2019 | VOL. 62 | NO. 12 (2019)..
9. Rodríguez N., Atencio H. et al: Interoperabilidad de funciones en el Modelo de Programación de Serverless Computing. In: IV CICCSI. Universidad Champagnat (2020).
10. EMR: Global Serverless Computing Market Outlook
11. https://www.expertmarketresearch.com/reports/serverless-computing-market (2021).
12. Ping-Min Lin, Alex Glikson : Mitigating Cold Starts In Serverless Platforms A Pool-Based Approach. https://arxiv.org/pdf/1903.12221.pdf
13. SAND: Towards high-performance serverless computing. In Proceedings of the 2018 USENIX Annual Technical Conference. 923ś935.

14. James Cadden, Thomas Unger, Yara Awad, Han Dong, Orran Krieger, and Jonathan Appavoo. 2020. SEUSS: skip redundant paths to make serverless fast. In Proceedings of the 15th European Conference on Computer Systems. 1ś15.
15. Edward Oakes, Leon Yang, Dennis Zhou, Kevin Houck, Tyler Harter, Andrea Arpaci-Dusseau, and Remzi Arpaci-Dusseau. 2018. SOCK: Rapid task provisioning with serverless-optimized containers. In Proceedings of the 2018 USENIX Annual Technical Conference. USENIX Association, 57ś70.
16. Kai-Ting Amy Wang, Rayson Ho, and Peng Wu. 2019: Replayable execution optimized for page sharing for a managed runtime environment. In Proceedings of the Fourteenth EuroSys Conference 2019. 1ś16.
17. P. Vahidinia, B. Farahani and F. S. Aliee: Cold Start in Serverless Computing: Current Trends and Mitigation Strategies, In: 2020 International Conference on Omni-layer Intelligent Systems (COINS), Barcelona, Spain, 2020, pp. 1-7, doi: 10.1109/COINS49042.2020.9191377.
18. Pawel Zuk and Krzysztof Rzadca. 2020: Scheduling methods to reduce response latency of function as a service. In Proceedings of the 2020 IEEE 32nd International Symposium on Computer Architecture and High Performance Computing. 132ś140
19. Artillery (s.f.) Artillery Docs [Online]. https://www.artillery.io/docs/guides/getting-started/core-concepts

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Detección de Vulnerabilidades en Smart Contracts Usando Machine Learning a Nivel de Bytecode

Matias A. Carballo, Hernan D. Merlino,

Laboratorio de Sistemas de Información Avanzados,
Facultad de Ingenieria, Universidad de Buenos Aires,
Paseo Colon 850, CABA, Argentina,
{mcarballo, hmerlino}@fi.uba.ar

Abstract. La seguridad de los contratos inteligentes es un aspecto crucial para garantizar su despliegue exitoso. En este estudio, presentamos una investigación que se centra en el desarrollo de técnicas de extracción de propiedades y preprocesamiento de información de contratos inteligentes, con el objetivo de entrenar nuevas herramientas basadas en aprendizaje de máquina. Además, proponemos diversas arquitecturas y configuraciones de redes neuronales, junto con modelos de lenguajes específicos, para llevar a cabo la detección de vulnerabilidades en estos contratos.

Keywords: Smart Contracts; Blockchain; Ethereum; Security; Solidity; Deeplearning.

1. Introducción

El uso de Smart Contracts[1] en la plataforma Ethereum[2] ha experimentado un notable aumento en los últimos años[3]. Estos contratos inteligentes presentan características únicas, como su naturaleza inmutable al ser desplegados en la red, la visibilidad pública del código compilado en la cadena de bloques y la capacidad de interactuar automáticamente con otros contratos según los términos acordados. Sin embargo, estas características también implican riesgos significativos, lo que requiere una exhaustiva revisión y evaluación de los contratos antes de su despliegue en la red.

A pesar de los esfuerzos de evaluación previa, existe la posibilidad de que los contratos sean desplegados en la red con comportamientos inesperados bajo ciertas condiciones, lo cual representa un riesgo para los usuarios. Se ha documentado un caso en 2016 en el que un cibercriminal logró robar 50 millones de dólares mediante un ataque de reentrada, que fue posible debido al orden de las instrucciones en el contrato[4].

En este trabajo de investigación, se analizan diversas estrategias para desarrollar herramientas de detección de vulnerabilidades en Smart Contracts, basadas en el análisis del código binario publicado en la red de Ethereum. Se examinan las diferentes partes que componen el código binario y se identifican las características relevantes que se pueden utilizar para detectar posibles vulnerabilidades. Además, se propone el uso de modelos de redes neuronales con diversas arquitecturas para evaluar la calidad de los resultados obtenidos mediante el análisis del código binario.

Mediante este enfoque, se busca mejorar la seguridad de los Smart Contracts en Ethereum y reducir el riesgo asociado a la explotación de vulnerabilidades. Los

resultados de esta investigación contribuirán al desarrollo de herramientas más efectivas para identificar y prevenir posibles ataques y comportamientos maliciosos en los contratos inteligentes.

2. Background

2.1 Contratos Intelgentes

Cuando se despliega un contrato en la red Ethereum, el código subyacente se representa en forma de bytecode, que es una forma compacta y binaria de representar las instrucciones y la lógica del contrato. El bytecode es una secuencia de instrucciones de bajo nivel que está diseñada para ser ejecutada por la Máquina Virtual Ethereum (EVM, por sus siglas en inglés), que es el entorno de ejecución en la red Ethereum.

El código fuente original del contrato, escrito en Solidity u otro lenguaje de programación compatible con Ethereum, se somete a un proceso de compilación para convertirlo en bytecode.

2.2 Extracción de Features

El bytecode puede ser dividido en cinco bloques de bytes que desempeñan funciones específicas. Estos bloques son: Creation Code, Function Selector, Wrapper, Body y Metadata Hash[5].

En el contexto del análisis que se llevará a cabo, se detectará la porción de bytecode que corresponde al Body de cada contrato. Esta porción del bytecode contendrá la información relevante que se utilizará en la investigación. Los otros bloques de bytes, como Creation Code, Function Selector, Wrapper y Metadata Hash, no serán considerados en este estudio y no estarán disponibles para los modelos utilizados. Esto permitirá a los modelos centrarse en la información esencial.



Fig. 1. La figura muestra la transformación de una cadena de bytes a una secuencia de operadores. Todos los contratos se truncaron a un largo máximo de 4048 operadores.

Otra consideración importante es que en la EVM, todas las instrucciones están compuestas por un operador, excepto las operaciones PUSH, que están seguidas de un valor inmediato. Sin embargo, para nuestra contextualización, descartaremos estos valores inmediatos, obteniendo una secuencia de operaciones (OpCodes)[6] comprensibles por la EVM.

Esta representación de secuencia de operaciones es útil para comprender la lógica y el flujo de ejecución del contrato, ya que permite analizar cada operación individualmente y entender cómo interactúan con la pila de la EVM.

2.3 Tipos de Vulnerabilidades

Se utilizarán como fundamento los hallazgos obtenidos en el análisis de 47.518 contratos[7] de Ethereum. Nuestro enfoque consiste en la aplicación de técnicas de aprendizaje automático a partir de bytecode presente en la red de Ethereum, con el objetivo de identificar patrones en los contratos vulnerables. En las fases de entrenamiento y predicción, se excluirán los archivos Solidity que contengan más de un contrato.

Abordaremos individualmente cada vulnerabilidad de forma separada. El objetivo de cada modelo será detectar si una vulnerabilidad específica está presente o no en un contrato dado. Por lo tanto, terminaremos con siete modelos diferentes de clasificación binaria, cada uno enfocado en una técnica de detección de vulnerabilidades particular.

Tabla 1. Tipos de vulnerabilidades y cantidad de contratos con las mismas detectadas.

Vulnerabilidad	Descripción	Cantidad de contratos
Access Control	Fallo al usar modificadores de funciones o el uso de tx.origin.	585
Arithmetic	Desbordamiento/subdesbordamiento de enteros.	2,137
Denial Service	Contrato abrumado por cálculos que consumen mucho tiempo.	644
Front Running	Dos transacciones dependientes que invocan el mismo contrato se incluyen en un bloque.	932
Reentrancy	Las llamadas de función reentrante hacen que un contrato se comporte de una manera inesperada.	1,003
Time Manipulation	La marca de tiempo del bloque es manipulada por el minero.	271
Unchecked Low Calls	call(), callcode(), delegatecall() o send() fallan y no se verifican.	479

Para lograr esto, balancearemos los conjuntos de datos de manera que haya una cantidad igual de contratos sin vulnerabilidades detectadas y contratos que presenten diferentes vulnerabilidades. Este enfoque nos permitirá entrenar y evaluar nuestros modelos de manera equilibrada, asegurando una representación adecuada de los casos con y sin vulnerabilidades.

3. Técnicas Propuestas

Dentro de esta sección se mostrará el detalle de cómo se aplican diferentes técnicas a los mismos set de datos confeccionados anteriormente de manera de obtener resultados comparables entre sí. Cada modelo utilizará las secuencias de *OPCODES* como entrada y la categoría binaria de presencia o ausencia de la vulnerabilidad como salida.

3.1 Regresión Logística e Ingeniería de Características

Este modelo se presenta como un punto de referencia inicial, con requisitos computacionales mínimos, un tiempo de entrenamiento rápido pero que aún sea lo suficientemente complejo como para obtener resultados significativos en la detección de vulnerabilidades.

Al aplicar TF-IDF a las operaciones, bigramas y trigramas de las secuencias de *OPCODES*, se asignará un peso a cada característica según su frecuencia en un contrato específico y su rareza en el conjunto de contratos. Esto permitirá capturar las operaciones y combinaciones de operaciones que son más relevantes y distintivas para cada contrato.

Se entrenará un modelo supervisado de regresión logística binaria para cada tipo de vulnerabilidad. La regresión logística binaria es un modelo estadístico que se utiliza para predecir una variable categórica binaria, en este caso, la presencia o ausencia de una vulnerabilidad específica.

La función logística en una dimensión tiene la forma:

$$P(Y = 1|\, X) = (1 + e^{-z})^{-1} \qquad (1)$$

Donde P(Y=1|X) es la probabilidad de que la vulnerabilidad esté presente dadas las características X, y z es una combinación lineal de los features ponderados por los coeficientes del modelo.

Para ajustar el modelo al conjunto de datos, se utiliza la función de pérdida de "negative log-likelihood" (log-verosimilitud negativa), que es una medida de qué tan bien se ajusta el modelo a los datos observados. Esta función de pérdida se utiliza para maximizar la probabilidad conjunta de los datos de entrenamiento dado el modelo.

3.2 Redes Neuronales Recurrentes

Las redes neuronales recurrentes se destacan por su habilidad de retener información dentro de una serie ordenada de datos, un ejemplo de estas es la red LSTM[8].

En particular se utilizó una arquitectura basada en LSTM[9]. A esta arquitectura se la complementará con dos capas densas al final, donde cada neurona de una capa se conecta con todas las de la siguiente, para realizar una tarea de clasificación.

Para que el modelo pueda utilizar una secuencia de operaciones, es necesario aplicar un proceso de tokenización. La tokenización implica tomar cada operación individual y transformarla en un token, que es una representación numérica única.

Como técnica de Tokenización se utilizó SentencePiece[10] que proporciona flexibilidad para ajustar el tamaño del vocabulario y la granularidad de las unidades de

subpalabras, así como también ofrece tokens especiales que ayudan a condensar información.

3.3 Modelo de Lenguaje y Transferencia de Conocimiento

Una metodología innovadora en el campo de las redes neuronales consiste en la transferencia de conocimientos entre ellas. Esta implica adaptar y aplicar el conocimiento aprendido de una tarea previa a una nueva tarea. Esto se logra mediante la transferencia de los pesos y parámetros del modelo previamente entrenado al nuevo modelo, lo que permite aprovechar la información relevante ya aprendida y acelerar el proceso de entrenamiento en el nuevo contexto. La transferencia de conocimiento puede mejorar el rendimiento y la eficiencia del modelo en la tarea objetivo.

Partiremos con el entrenamiento de un modelo de lenguaje utilizando una arquitectura encoder-encoder de manera tal que el encoder tenga la misma arquitectura basada en LSTM[9] para poder luego hacer dicha transferencia de conocimiento.

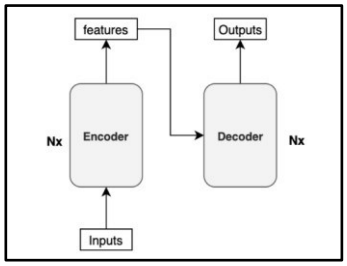


Fig. 2. Arquitectura encoder-decoder. Esta figura muestra los componentes esenciales de la arquitectura. Se deja en evidencia la relación de las distintas partes con la información de entrada y salida, así como también la ubicación de la representación vectorial.

Dentro de la arquitectura de encoder-decoder, el encoder es responsable de capturar la información intrínseca de las secuencias de entrada y generar una representación vectorial que capture su significado. El decoder utiliza dicha representación para contextualizar y generar elementos secuenciales de manera coherente.

Para entrenar un modelo de lenguaje con esta arquitectura, la información de entrada consiste en una lista de tokens. La información de salida es la misma lista de tokens, pero con un corrimiento, lo que implica que cada token en la salida es el siguiente token en la secuencia original.

Tabla 2. Ejemplo de set de datos para el entrenamiento del modelo de lenguaje.

Información de entrada	Información de salida
xxbos jumpdest push1 push1 swap1	jumpdest push1 push1 swap1 sload
exp dup2 sload dup2 push20	dup2 sload dup2 push20 mul

Para la confección de este dataset no es necesario el conocimiento previo de la presencia o no de vulnerabilidades, lo que nos permite utilizar cualquier contrato desplegado en la red de ethereum.

Entrenaremos el modelo de lenguaje utilizando la totalidad de los contratos disponibles, este modelo será el mismo para todas las vulnerabilidades.

Para el proceso de transferencia de conocimiento tomaremos el encoder previamente entrenado y lo complementaremos con dos capas de redes neuronales densas para especializarse en la tarea de clasificación. Una diferencia a la hora del entrenamiento es la técnica de congelamiento, esta consiste en una primera iteración del entrenamiento donde los pesos del encoder no se actualizarán según el gradiente, y se utilizara un parámetro de aprendizaje mayor. En la segunda iteración se reduce ese coeficiente de aprendizaje y se entrena toda la arquitectura en conjunto.

3.4 Transformers

Debido a lo extenso que puede ser un contrato, propondremos una arquitectura cuyo núcleo contiene una arquitectura de Transformers[11]. Por otra parte, entrenaremos un modelo de lenguajes que se ajuste a este núcleo para poder aplicar transferencia de conocimiento.

3.5 RoBERTa

Para el núcleo de la arquitectura utilizaremos una implementación basada en RoBERTa[12], la cual adaptaremos con la siguiente configuración: 512 tokens, 4 hidden layers, 6 attention heads y 216 embedding size.

3.6 Tokenización y Modelo de Lenguaje

Para la tokenización realizaremos un byte-level Byte-Pair-Encoding[13], esta puede capturar subunidades significativas en la secuencia, lo que ayuda a reducir la dimensionalidad creando una representación más eficiente y mejorar la compresión de datos.

Entrenaremos RoBERTa[12] como modelo de lenguaje, esto se realiza mediante la técnica de máscaras, una máscara es un token específico que reemplaza un 15% de los tokens presentes en cada secuencia. El objetivo del entrenamiento es lograr predecir que tokens debe reemplazar en las máscaras para obtener la secuencia original.

3.7 Arquitectura

Propondremos el uso de una arquitectura siamesa de 8 bloques de RoBERTa[12], seguida de dos capas de redes neuronales densas. Esta elección se debe a que nuestro corpus de texto consta de 4096 tokens, mientras que el límite de tokens de entrada del modelo es de solo 512.

En esta arquitectura siamesa, dividimos la información de entrada en 8 bloques, lo que nos permite procesar cada bloque de forma independiente. Cada bloque se pasa a

través de un bloque correspondiente de RoBERTa[12], que comparte los mismos pesos y arquitectura. Esto significa que los 8 bloques de RoBERTa[12] son idénticos, lo que facilita la paralelización del procesamiento.

Después de procesar cada bloque individualmente, recopilamos la información mediante dos capas de redes neuronales densas. Estas capas se utilizan para fusionar la información extraída de los 8 bloques y generar una representación global del texto de entrada.

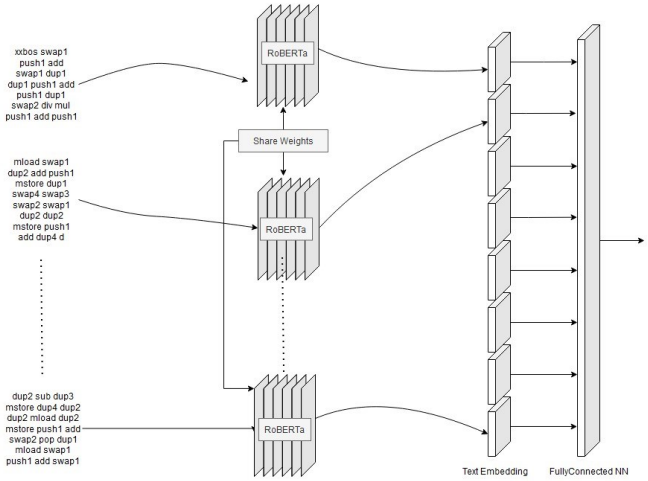


Fig. 3. La figura muestra una representación de la arquitectura siamesa.

4. Resultados

Para poder evaluar y comparar el rendimiento de los diferentes modelos en relación a los distintos tipos de vulnerabilidades, utilizamos la métrica del área bajo la curva ROC (Receiver Operating Characteristic).

La curva ROC es una representación gráfica que muestra la relación entre la tasa de verdaderos positivos (TPR) y la tasa de falsos positivos (FPR) a medida que se varía el umbral de clasificación del modelo. El área bajo esta curva (AUC-ROC) es una medida numérica que resume el rendimiento global del modelo. Un AUC-ROC mayor indica un mejor rendimiento, ya que significa que el modelo tiene una mejor capacidad para distinguir entre las clases positivas y negativas.

Tabla 3. Los resultados del Area bajo la curva ROC para cada modelo.

Vulnerabilidad	Reg. Log.	LSTM	LSTM + LM	Transformers	Transf. + LM
Acc. Ctrl.	**0.87**	0.57	**0.87**	0.61	0.76
Arithmetic	0.90	0.90	**0.91**	0.68	0.82
D. Service	0.87	0.62	**0.91**	0.60	0.82
F. Running	0.86	0.86	**0.89**	0.53	0.78

Reentrancy	0.83	0.51	**0.91**	0.62	0.81
Time Man.	0.81	0.79	**0.85**	0.53	0.68
Unchecked L.C.	0.82	0.53	**0.83**	0.55	0.71

Observamos que los modelos basados en redes neuronales recurrentes que se inicializan con un modelo de lenguaje pre-entrenado tienden a mostrar mejores resultados en comparación con el resto de los enfoques. Así mismo, cuando analizamos el modelo de RoBERTa[12] inicializado con un modelo de lenguaje pre-entrenado, también arroja una mejora sustancial sobre el mismo modelo inicializado aleatoriamente.

Por otra parte, la regresión logística muestra un alto rendimiento, lo que sugiere grandes oportunidades de mejora para arquitectura de Transformers propuesta, ya sea en volumen de datos para el entrenamiento, o en diferentes configuraciones de la misma.

5. Conclusiones

Hemos logrado entrenar varios modelos que son capaces de identificar vulnerabilidades con diferentes niveles de precisión. En segundo lugar, hemos aplicado técnicas de preprocesamiento y segmentación de datos para evitar la inclusión de información redundante e innecesaria.

El enfoque utilizado en esta investigación presenta la ventaja de ser flexible y adaptable a medida que surjan nuevos tipos de vulnerabilidades. Esto se debe a las técnicas aplicadas y la capacidad de los modelos para aprender y generalizar a partir de los datos de entrenamiento.

Finalmente es importante destacar que las técnicas utilizadas en este enfoque se centran en el análisis de secuencias de operaciones para identificar patrones relevantes. Esto proporciona una ventaja significativa en términos de velocidad de respuesta en comparación con las técnicas tradicionales que implican pruebas exhaustivas o métodos computacionalmente costosos y lentos.

Agradecimientos

Este proyecto forma parte del PIDAE 2022 (1-285) Computación Descentralizada y Silos de Procesamiento.

References

1. Szabo, N.: Smart contracts: building blocks for digital markets, EXTROPY: The Journal of Transhumanist Thought, (16), vol. 18, no. 2, 1996.
2. Buterin, V.: et al., A next-generation smart contract and decentralized application platform, white paper, vol. 3, no. 37, 2014

3. Asmakov, A.: Ethereum Smart Contracts Deployment Jumped 293% in 2022: Alchemy Developer Report https://decrypt.co/119371/ethereum-smart-contracts-deploymentjumped-293-2022-alchemy-developer-report
4. Meyer, B.: Ethereum smart contract vulnerabilities can lead to millions in losses https://cybernews.com/security/ethereum-smart-contract-vulnerabilities/
5. Santander, A., Arias, L.: Deconstructing a Solidity Contract, https://blog.openzeppelin.com/deconstructing-a-solidity-contract-part-i-introduction832efd2d7737/
6. Wood, G.: Ethereum: A secure decentralised generalised transaction ledger berlin version beacfbd – 2022-10-24
7. Durieux, T., Ferreria, J.F., Abreu, R., Cruz, P.: Empirical Review of Automated Analysis Tools on 47,587 Ethereum Smart Contracts, arXiv:1910.10601
8. Hochreiter, S., Schmidhuber, J.,: Long Short-Term Memory, 1997, https://www.bioinf.jku.at/publications/older/2604.pdf
9. Merity, S., Keskar, N. S., Socher, R.: Regularizing and Optimizing LSTM Language Models, arXiv:1708.02182
10. Kudo, T., Richardson, J.,: SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing, arXiv:1808.06226
11. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., Polosukhin, I.,: Attention Is All You Need, arXiv:1706.03762
12. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: RoBERTa: A Robustly Optimized BERT Pretraining Approach, arXiv:1907.11692
13. Sennrich, R., Haddow, B., Birch, A.: Neural Machine Translation of Rare Words with Subword Units, arXiv:1508.07909

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Implementando estrategias de resiliencia en una arquitectura basada en microservicios

Sergio Leonel Suárez[1,2], Enzo Rucci[3][0000-0001-6736-7358],
Victor Betran[2], and Diego Montezanti[3][0000-0003-0436-0997]

1 Facultad de Informática, UNLP. La Plata (1900), Bs As, Argentina
[2] PedidosYa
{sergio.suarez,victor.betran}@pedidosya.com
3 III-LIDI, Facultad de Informática, UNLP – CIC. La Plata (1900), Bs As, Argentina
{erucci,dmontezanti}@lidi.info.unlp.edu.ar

Resumen En los últimos años, se ha incrementado la adopción de arquitecturas de microservicios para ayudar a superar las limitaciones de los sistemas monolíticos. En los sistemas basados en microservicios, la resiliencia es crucial debido al impacto directo de las fallas en el negocio de una empresa. Debido a esto, han surgido los patrones de diseño para resiliencia como estrategias para gestionar las fallas y mitigar sus efectos negativos. En este trabajo se analiza el comportamiento de algunos patrones para resiliencia dentro del ecosistema de microservicios de PedidosYa, centrándose en el microservicio *Niles*. Para ello, se estudiaron diversos escenarios de fallas y la aplicación de patrones para darles solución. Además, se obtuvieron resultados experimentales que permitieron evaluar el impacto de la aplicación de estos patrones, evidenciando así que *Niles* ha logrado convertirse en uno de los componentes más robustos y utilizados en PedidosYa.

Keywords: Arquitecturas de Microservicios · Resiliencia · Sistemas Monolíticos · Fallos · Patrones de diseño · Sistemas distribuidos

1. Introducción

El término *Arquitectura de microservicios* ha surgido en los últimos años para describir una manera particular de diseñar aplicaciones como conjuntos de servicios desplegables de forma independiente, de manera de dar respuesta a las limitaciones de los sistemas monolíticos [6]. Si bien no existe una definición precisa de este estilo arquitectónico, existen características comunes en torno a la organización, la capacidad empresarial, la implementación automatizada y el control descentralizado de lenguajes y datos [1].

Desafortunadamente, estas arquitecturas también presentan una serie de desventajas. Por ejemplo, cuando ocurre un fallo, resultan afectados tanto el microservicio que lo experimenta como los otros que dependen de él. Otras desventajas son: la complejidad de los sistemas distribuidos, la ausencia de un método específico para la descomposición en microservicios de un sistema de grandes dimensiones, la necesidad de mantener la consistencia, la dificultad en la monitorización del estado de los componentes del ecosistema y la gestión de la seguridad.

La resiliencia es uno de los aspectos no funcionales más buscados, especialmente en los sistemas de gran escala. El manejo de las fallas es una cuestión fundamental, ya

que su impacto tiene relación directa con el negocio de una empresa, traduciendo en pérdidas económicas el tiempo de no disponibilidad de los servicios [3]. Es por ello que entran en juego los patrones de diseño para resiliencia entre microservicios, que son estrategias que ayuden a lidiar con estas fallas y mitigar sus consecuencias negativas.

Este trabajo realiza un análisis de la resiliencia en el ecosistema de microservicios de PedidosYa, una compañía de *delivery* en línea que opera en más de 10 países de América Latina. Esta empresa cuenta con una arquitectura basada mayormente en microservicios, la cual procesa aproximadamente 4 millones de órdenes por semana. El análisis propuesto contempla la aplicación de algunos patrones de resiliencia y las mejoras derivadas de ello, en cuanto a conversiones de negocio y otras métricas.

El resto del artículo se organiza de la siguiente forma. La Sección 2 brinda el marco referencial para el trabajo. La Sección 3 describe los patrones utilizados. La Sección 4 muestra algunos resultados experimentales. Por último, la Sección 5 resume las conclusiones y posibles líneas de trabajo futuro.

2. Marco Referencial

2.1 Fallos en arquitecturas de microservicios

En una arquitectura basada en microservicios existen situaciones que pueden causar que el sistema adquiera un comportamiento inestable. Si bien algunas son comunes a todos los sistemas de software (como una base de datos inconsistente o un recurso agotado), existen casos que son propios de los sistemas distribuidos [5], como son:

- Errores o lentitud en la red: debido a que la interacción entre múltiples servicios es muy frecuente, ante un error de red, un microcorte, el daño de un componente (como el servidor de DNS), o la congestión, el ecosistema resulta afectado.
- Picos de tráfico: dado que cada microservicio es autónomo e independiente, es posible definir sus límites de operación mediante parámetros como la cantidad de requerimientos por minuto que es capaz de soportar. Si un servicio recibe más tráfico del esperado, es posible que se degrade, comenzando a funcionar con lentitud o de forma anómala.
- Priorización incorrecta: no todos los servicios dentro de una compañía tienen el mismo nivel de prioridad. Por ejemplo, un servicio que maneja transacciones monetarias tiene naturalmente una prioridad mayor que un servicio que se encarga de la administración de comentarios. Si las prioridades no están correctamente definidas, el fallo de cualquier servicio podría tener impacto similar. El manejo de prioridades y de errores debe ser definido individualmente sobre cada servicio, de manera de no penalizar a todo el ecosistema ante un fallo de cualquiera de sus elementos.

La resiliencia es una característica que debe ser concebida desde el propio diseño de la arquitectura mediante un correcto modelado de los microservicios. En este contexto, la aplicación de patrones ayuda a construir una arquitectura de microservicios resiliente frente a los fallos.

2.2 Arquitectura de microservicios en PedidosYa y caso de estudio: servicio *Niles*

En cuanto al negocio, la empresa se encuentra dividida en lo que se conocen como verticales, tales como Restaurantes, Farmacias, Mercados y Bebidas. En la vertical de Restaurantes, que es la que más ganancias genera a la empresa, toda la lógica se encuentra dispersa en microservicios, siendo *Niles* uno de los más importantes. *Niles* se encarga de proporcionar al usuario el menú de un restaurante. Realiza múltiples consultas a otros microservicios con el fin de generar el menú, compuesto por secciones con productos. Además, aporta información básica para cada uno de ellos, como su nombre, descripción, imagen, precio y descuentos aplicables, y brinda datos adicionales tales como su popularidad de venta, si es un producto recomendado o si fue marcado como favorito.

Los módulos de visualización dentro de cada aplicación móvil, denominados *shopDetail*, invocan a *Niles* para consultar el menú. El proceso de retorno de un menú es el siguiente: (1) *Niles* recibe el requerimiento de menú para un restaurante; (2) verifica la existencia del menú para el restaurante solicitado en una caché distribuida; (3) en caso de que el menú no exista en la caché, se consultan diferentes servicios de modo de completar la información requerida. Una vez que se obtiene el menú completo, se persiste en la caché para próximas solicitudes; (4) en cualquiera de los casos anteriores, el menú se procesa aplicando diferentes filtros (validaciones de edad, stock disponible, ventanas de horario correctas) y ordenamientos. El microservicio cuenta con una caché centralizada, compartida por múltiples instancias de *Niles*, y una caché de tipo local para guardar información que rara vez cambia, como los países y las categorías de productos. Para poder presentar un menú actualizado, *Niles* recibe eventos de novedades desde otros microservicios tales como *Items-service* y *Battlefront*. Al momento de recibir una actualización, *Niles* elimina de la caché la entrada correspondiente a ese restaurante, el cuál será ingresado nuevamente cuando reciba un requerimiento desde las aplicaciones móviles. El flujo completo de comunicación entre todos los componentes se ilustra en la Figura 1.

2.3 Microservicios asociados

Para brindar su servicio de forma completa, *Niles* interactúa con varios microservicios, que se describen en detalle en [8]. Sin embargo, para realizar los experimentos que permitieron evaluar el impacto de la aplicación de los patrones, únicamente se afectaron las interacciones de *Niles* con dos de ellos, cuyas funcionalidades se describen a continuación:

- *Items-service*: es el microservicio que alimenta al menú. Se encarga de proveer las secciones y productos, así como también sus opciones de configuración. Para cada sección, producto u opciones de productos, administra datos como el nombre, imagen, descripción y precio. Ante cualquier cambio de los datos en un ítem, el servicio envía novedades (mediante tópicos) que llegan a las colas de mensajes de *Niles*, que es quien se encarga de actualizar la caché centralizada.

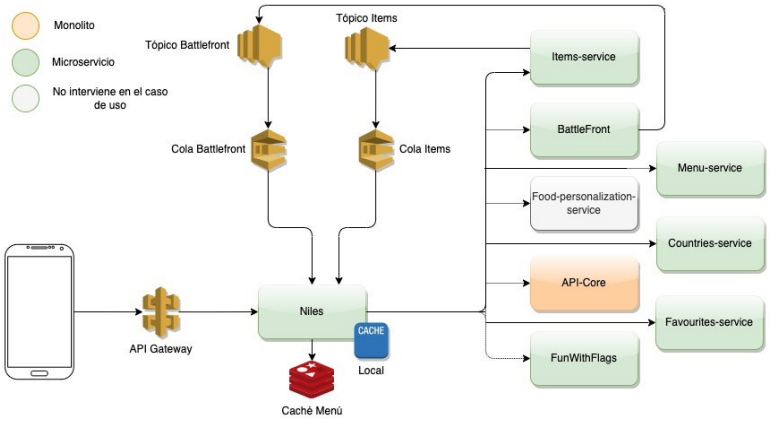


Figura1: Interacción de *Niles* con sus componentes asociados

En caso de que *Items-service* no funcione correctamente (debido a una caída temporal o incidente serio), *Niles* no podría retornar el menú. Si bien la información de *Items-service* se mantiene en una caché centralizada, la misma cuenta con un tiempo de vida limitado por lo que, ante una falla, podría mostrarse un menú desactualizado.

- *Favourites-service*: es el servicio que provee la información de todas las entidades que un usuario marca como sus preferidas. Maneja entidades como restaurantes, productos o configuraciones. *Niles* consume *Favourites-service* para poder agregar o remover productos a la lista de favoritos.

3. Tratamiento de fallos en *Niles*

Un microservicio se implementa como un servidor que se encarga de recibir los requerimientos entrantes, empleando un *pool* de hilos para poder atenderlos. El número de requerimientos que se pueden atender simultáneamente está limitado por la cantidad de hilos disponibles en el *pool*; éstos se reutilizan para nuevos requerimientos a medida que se liberan. Como los microservicios son autónomos y se encargan de una funcionalidad concreta, es común que deban consultar información desde otras entidades externas, como bases de datos, cachés u otros microservicios. Cada vez que un recurso externo es consultado, el *thread* que realiza la consulta queda bloqueado en espera, por lo que no puede atender nuevas peticiones.

3.1 El patrón *Timeout*

En cualquier momento, la entidad consultada puede degradar sus tiempos de respuesta, debido a errores, lentitud en la comunicación o problemas internos; esta situación puede ocurrir frecuentemente. En el caso de que no se establezca un límite de espera para el servicio que consume al recurso que presenta inconvenientes, el hilo que atiende el requerimiento puede quedar bloqueado indefinidamente. Esta situación eventualmente puede desembocar en que todos los *threads* queden bloqueados, y que

incluso se empiecen a encolar peticiones a la espera de que se liberen hilos. Una posible solución consiste en la implementación del patrón *Timeout*, que establece un tiempo de espera máximo al momento de consumir recursos externos; una vez que el lapso se cumple, el servicio consumidor corta la comunicación, liberando a los hilos involucrados para que puedan atender nuevos requerimientos [2]. Un aspecto destacable es que la definición de los tiempos de espera depende de cada microservicio, por lo que se requiere conocer sus indicadores de nivel de rendimiento; (o analizar métricas tales como latencia, *error rate* o tráfico recibido[1], si esta información no está disponible).

Para la entrega de un menú por parte de *Niles*, la petición a *Items-service* es la más importante, ya que sin esa información no es posible recuperar secciones y productos; en tanto, la petición a *Favourites-service* es de relevancia menor, ya que se puede presentar un menú completo sin la información sobre productos marcados como favoritos. Ambas peticiones se ejecutan en paralelo para optimizar recursos y retornar el menú en menos tiempo.

En una situación normal, la solicitud de un menú desde *Niles* se resuelve en 50 ms[2] (ya que *Items-service* responde en 50 ms y *Favourites-service* en 10 ms). Sin embargo, si *Favourites-service* se degrada (y empieza a responder en tiempos altos), ocurren dos problemas. El primero es que, como ambos servicios se ejecutan en paralelo, la respuesta final se produce junto con la del servicio con mayor latencia (en este caso *Favourites-service*), por lo que se penaliza la entrega de un menú debido una petición de escasa relevancia. El segundo (y más grave) es que se pueden bloquear la totalidad de hilos. *Favourites-service* continúa encolando peticiones en espera, por lo que los tiempos de respuesta aumentan aún más. Si se conoce el tiempo de respuesta típico de *Favourites-service* (en este caso, alrededor de 10ms), se puede configurar por ejemplo una espera máxima de unos 15ms. De esta forma, no se penaliza excesivamente el menú (simplemente se entrega sin la información de favoritos) ni tampoco se bloquean hilos a la espera de una respuesta que probablemente nunca llegue.

3.2 El patrón *Retry*

En las arquitecturas distribuidas en general, donde diferentes servicios se comunican constantemente, ocurren con frecuencia errores transitorios a causa de pérdidas momentáneas de conexión a la red o no-disponibilidad temporal de recursos. Normalmente, estos errores se manifiestan durante lapsos cortos de manera de que, si el recurso vuelve a ser consultado a la brevedad, responde correctamente. Una posible solución es la aplicación del patrón *Retry*, que consiste simplemente en reintentar una operación que ha fallado. Según el tipo de error detectado y/o el número de intentos, se pueden realizar diversas acciones. Por ejemplo, si el código del error retornado indica que es un fallo temporal o atípico, la operación puede reintentarse de inmediato, ya que probablemente no vuelva a producirse el mismo error. En tanto, si el error se debió a

[1] Estas métricas son provistas por herramientas que monitorean el desempeño de un servicio, conocidas como APM (*Application Performance Monitoring*)

[2] Este valor se obtiene con la métrica p95, que es la latencia medida en forma de percentil. Una métrica p95 con valor 50ms significa que el 95% de los requerimientos se resuelven en un tiempo menor o igual a 50ms

un problema de conexión o a un pico de peticiones al servicio, es prudente esperar un tiempo antes de reintentar la operación. Finalmente, si el código de error indica que el fallo no es temporal, la operación debería ser cancelada, reportándolo para gestionarlo de manera adecuada. Estas acciones pueden combinarse para crear una política de reintentos ajustada a las necesidades de la aplicación particular.

Debido a que *Niles* no puede generar el menú si *Items-service* (en particular) no está operativo (o fallan peticiones hacia él), existe una fuerte dependencia entre ellos que genera fragilidad en la funcionalidad de *Niles* ante inconvenientes en la red. La aplicación de *Retry* en la comunicación entre ambos servicios permite solucionar estos errores transitorios.

Cabe destacar que, si bien ante un fallo de *Items-service* el usuario podría actualizar la pantalla del móvil, reiniciar la aplicación o cerrar sesión y volver a intentar obtener el menú, existen casos en los que no es posible esa interacción (como procesos *batch*, actualizaciones masivas, liquidaciones de sueldo, etc.), para los cuales el patrón *Retry* es de suma importancia. Por otra parte, es importante configurar y administrar correctamente los reintentos, aumentándolos sólo para los recursos que son vitales para el caso particular. Por ejemplo, no tiene sentido aplicar reintentos para recuperar un producto favorito o una reseña, ya que se penaliza a la funcionalidad principal. Tampoco deberían realizarse reintentos si el código de respuesta HTTP indica que el servidor está sobrecargado (para no saturarlo); o en el caso de una excepción interna (como un envío incorrecto de parámetros) que sea responsabilidad del cliente que realiza la petición.

3.3 El patron Circuit Breaker

El patrón *Circuit Breaker* está inspirado en el funcionamiento de un disyuntor eléctrico, el cual es capaz de interrumpir la corriente en un circuito cuando ocurren determinadas fallas, con el objetivo de proteger a las personas. De manera más abstracta, impide la destrucción total de un sistema ante una falla que afecta a una parte de él. El disyuntor puede restablecerse para restaurar la funcionalidad una vez pasado el riesgo. En el ámbito del software, este concepto se aplica para proteger operaciones relevantes o críticas, mediante un interruptor automático que evita realizarlas cuando el sistema no está en buenas condiciones (en lugar de reintentar) [2] [4] [7].

En el estado normal (cerrado), el disyuntor ejecuta las operaciones como es habitual; si una llamada falla, el disyuntor lo registra, y una vez que la cantidad o la frecuencia de las fallas excede un umbral, el disyuntor se activa y abre el circuito. Cuando el circuito está abierto, las llamadas al disyuntor fallan inmediatamente, por lo que no se intenta ninguna operación. Después de un cierto tiempo, el *Circuit Breaker* decide que la operación tiene cierta probabilidad de éxito y pasa al estado semi-abierto; en él, la siguiente llamada al disyuntor intenta ejecutar nuevamente la operación crítica. Si tiene éxito, el disyuntor se restablece al estado cerrado; pero si la llamada de prueba falla, vuelve al estado abierto hasta que transcurre otro tiempo de espera.

El disyuntor degrada automáticamente la funcionalidad cuando el sistema está bajo estrés. Esto puede impactar en el negocio de la empresa, por lo que es fundamental el criterio para el manejo de las llamadas que se realizan cuando el circuito está abierto. Dependiendo de los casos de uso del sistema, cada disyuntor implementado puede estar

configurado para detectar diferentes tipos de fallas y manejar diferentes parámetros (como, por ejemplo, distintos umbrales). Al igual que con el patrón *Retry*, la configuración debe ser cuidadosa; por ejemplo, si el cliente envía parámetros incorrectos, el *Circuit Breaker* debería permanecer cerrado para no evitar llamadas que tienen que ejecutarse.

Tanto *Niles* como *Items-service* son servicios fundamentales para el negocio, ya que su cáıda impacta directamente sobre la generación de órdenes, por lo cual deben presentar alta disponibilidad y funcionar correctamente en todo momento[3]. *Itemsservice* recibe internamente las novedades de modificaciones de ítems. Debido a que restaurantes grandes pueden producir actualizaciones masivas, las novedades no se dan siempre de manera controlada. Estas novedades actualizan la base de datos propia de *Items-service*, la cual utiliza la técnica de Throttling[4] para controlar estos picos de operaciones: básicamente, la base de datos advierte que se ha llegado al límite de peticiones y agrega dinámicamente mayor capacidad. Sin embargo, este proceso demanda cierto tiempo, durante el cual todas las peticiones realizadas fallan. Mientras tanto, *Niles* continúa solicitando información a *Items-service*, y el agregado de reintentos empeora la situación. Debido a esto, *Items-service* no logra recuperarse y termina afectando a los demás servicios que lo consumen.

La aplicación del *Circuit Breaker* evita saturar a *Items-service* con requerimientos desde *Niles*. Cuando el disyuntor entre ambos pasa a estado abierto, en principio, no podría mostrarse el menú, pero en general la respuesta se brinda desde la caché propia de *Niles*. Así, se evita abrumar a un servicio que ya llegó al límite de requerimientos, pero que estará nuevamente disponible en el corto plazo.

4. Resultados Experimentales

4.1 Diseño experimental

En general, en las comunicaciones entre microservicios, es buena práctica definir *Timeouts* adecuados, configurar reintentos en algunos casos e implementar un *Circuit Breaker* que evite saturar de requerimientos a los microservicios que son consumidos. Debido a esto, se seleccionaron estos patrones para analizar y evaluar el impacto de su aplicación en *Niles* y sus microservicios asociados[5]. Para ello, se realizaron dos procesos experimentales para cada uno de ellos (se puede encontrar un mayor nivel de detalle sobre el diseño experimental en [8]):

1. El primer proceso consistió en enviar a *Niles* una serie de peticiones HTTP para emular el funcionamiento normal. Luego de un determinado tiempo, se sometió a *Niles* a una fuerte ráfaga de peticiones para degradar su funcionamiento. Estas peticiones se realizaron sobre una versión de *Niles* que no incluía la aplicación de

[3] Dentro de la compañía, estos servicios se catalogan como Tier1

[4] https://aws.amazon.com/es/premiumsupport/knowledge-center/dynamodb-tablethrottled

[5] Dentro de PedidosYa, los patrones Timeout, Retry y Circuit Breaker están implementados en una librería interna denominada peya-kator-utils. Los detalles vinculados a la implementación pueden encontrarse en [8]

ningún patrón.

2. El segundo proceso replicó al anterior, pero en este caso sobre una versión de *Niles* que incluía la aplicación del patrón de interés. Para ambos casos se monitorizó el funcionamiento de *Niles*, para luego analizar los datos obtenidos. De esta manera, se pudieron cuantificar los beneficios del uso de cada patrón.

A continuación, se describen varios aspectos vinculados al trabajo experimental.

Instalación de *Niles* utilizando Jarvis, una herramienta de la compañía que permite múltiples operaciones, como la creación y administración de nuevos proyectos y servicios en diferentes lenguajes, librerías, diferentes tipos de bases de datos, tópicos, colas de mensajes, cachés, programación de *cron jobs*, etc.

Ejecución de una serie de peticiones HTTP sobre uno o varios recursos dentro de *Niles*. Mediante Jarvis, se programa una tarea que se lanza a demanda y ejecuta un *script* de la herramienta K6, que permite realizar pruebas de performance. K6 se configura para que, durante el primer minuto de ejecución, intente realizar 200 peticiones (dependiendo de los recursos disponibles en el servidor subyacente); durante el segundo minuto, aumenta para intentar llegar a 500 requerimientos; y, durante los últimos 10 minutos se realizan aproximadamente 700 requerimientos por minuto. Por lo tanto, la duración total de la prueba es de 12 minutos.

Ejecución de fuertes ráfagas de peticiones para degradar un servicio consumido por *Niles*. Para esto se utiliza *Autocannon*, una herramienta de propósito similar a K6. Esta herramienta es más fácil de usar, requiere menos configuraciones y se puede ejecutar localmente, sin utilizar recursos de la compañía. *Autocannon* se configura para realizar 800 peticiones durante 300 segundos al servicio de favoritos de *Niles*, luego de 5 minutos de iniciadas las pruebas con K6. Al enviar una fuerte ráfaga de requerimientos en un período corto (y no hacerlo gradualmente), se evita que el servicio escale horizontalmente, lo que interferiría con el propósito del experimento.

Evaluación del impacto del patrón mediante el análisis de las mejoras generadas en base a su aplicación. Para esto se utilizó la herramienta *DataDog*, la cual ofrece métricas de rendimiento y posee integración con diferentes elementos de infraestructura, como bases de datos, logs y cachés.

4.2 Experimentos y Resultados Obtenidos para *Timeout*

Este experimento se realizó para demostrar cómo, al no configurar correctamente los tiempos de espera en las peticiones HTTP, el menú se ve penalizado por los tiempos altos de respuesta de un servicio de menor prioridad como es *Favouritesservice*. A partir de una instalación de *Niles* sin ningún patrón, se inicia (mediante K6) la realización de peticiones sobre el *endpoint* que ofrece el menú. Luego de aproximadamente cinco minutos, se procede a enviar una ráfaga de peticiones hacia *Favourites-service*.

En la parte izquierda de la Fig. 2 se ve como el tráfico no sigue un patrón regular (alterna periodos de pocas y excesivas peticiones), lo que evidencia que existen hilos bloqueados en espera en determinados momentos. En tanto, a la derecha se observa cómo los tiempos del menú se ven afectados, ya que existen peticiones en espera desde *Favourites-service*.

A continuación, se repite el experimento, pero incorporando la configuración de un límite de espera en cada petición HTTP saliente desde *Niles*. Por lo tanto, cuando *Favourites-service* sufre una ráfaga de requerimientos y se degrada, no se penaliza al menú. A la izquierda de la Fig. 3 se puede observar cómo el tráfico entrante aumenta de forma estable, sin alternar períodos de alto y bajo tráfico. En tanto, a la derecha, se puede ver que el tiempo de latencia no supera los 170 ms (salvo en dos picos), lo que resulta mucho menor a los 6000 ms en promedio sin la aplicación del patrón *Timeout*. Esta mejora en el rendimiento se debe a que se finaliza intencionalmente la comunicación con las dependencias luego del lapso establecido.

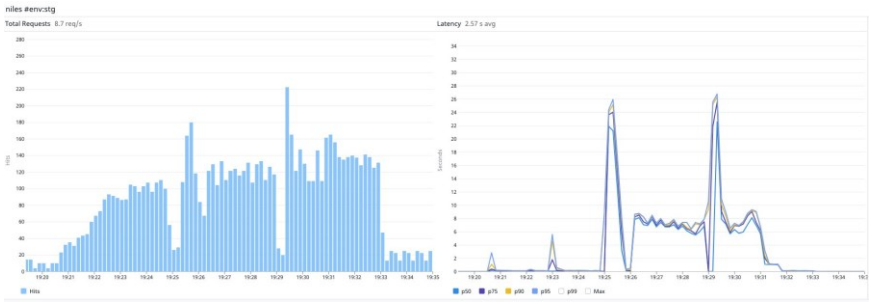


Figura2: Tráfico entrante (izq) y tiempos de respuesta de *Niles* (der)

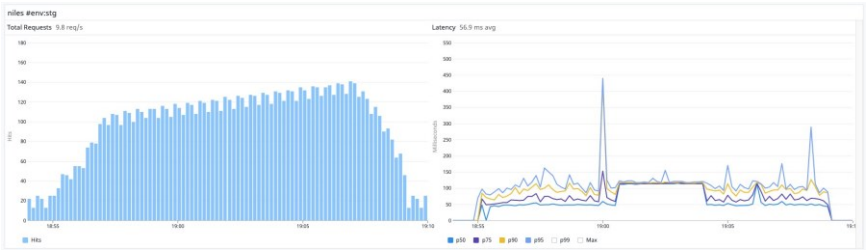


Figura3: Tráfico entrante (izq) y tiempos de respuesta de *Niles* (der) con aplicación de *Timeout*

4.3 Experimentos y Resultados Obtenidos para *Retry*

Para la realización de las pruebas del patrón *Retry* no se ejecutan ráfagas de peticiones sobre *Favourites-service*, pero se realiza una modificación en su código para simular el fallo en una de cada diez peticiones. El objetivo es evaluar el comportamiento de los servicios frente a errores temporales o lentitud en la red con la aplicación de reintentos.

A partir de una instalación de *Niles* sin la aplicación de ningún patrón, se ejecutan las peticiones de prueba utilizando K6. En la parte superior de la Fig. 4 se puede observar el tráfico recibido en *Niles*, donde los errores se marcan en color rojo. Sin embargo, haciendo foco en el tipo de errores (en la parte inferior) se puede ver que presentan código de estado 200, lo que significa que las peticiones a *Niles* finalizan exitosamente (se retorna el menú al usuario), pero en la ejecución ocurrió algún error

(que es el recibido desde *Favourites-service*. Debido a que este servicio no es obligatorio para generar el menú, no se penaliza al mismo.

Mediante el envío de parámetros al cliente que realiza la petición, se configura una cantidad de 3 reintentos por defecto en la comunicación desde *Niles* hacia *Favourites-service*. En la parte superior de la Fig. 5 se puede ver el tráfico recibido por la versión de *Niles* que hace uso del patrón *Retry* (con los errores marcados en rojo), mientras que la parte inferior se focaliza sobre los errores con código de respuesta 200: bien aún existen errores, la diferencia es notable cuando se aplica el patrón, ya que el contador que se aprecia en la Fig. 5 muestra una cantidad total de 54 errores, cuando este valor ascendía a 822 en ausencia de *Retry* (ver Fig. 4).

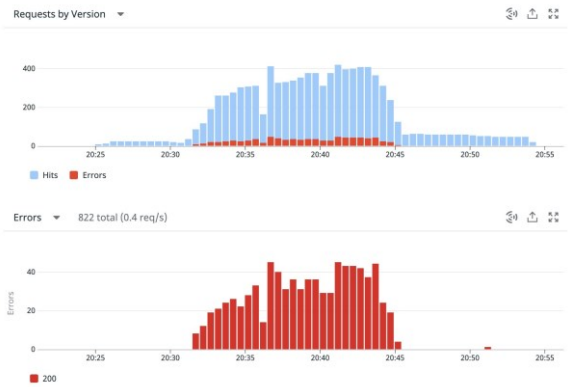


Figura4: Peticiones hacia *Niles* (arriba) y cantidad de errores con código de estado 200 (abajo)

4.4 Experimentos y Resultados Obtenidos para *Circuit Breaker*

El objetivo de estas pruebas es mostrar que la aplicación del patrón evita que el servicio consumidor utilice recursos en realizar peticiones que tienen muchas probabilidades de fallar. La realización de pruebas sigue la misma metodología explicada en la sección 4.1. En la Fig. 6 se puede observar el aumento de tráfico hacia *Niles* (en su versión sin la aplicación de ningún patrón) y los errores provocados intencionalmente en *Favourites-service* (marcados en rojo). En la parte superior, se refleja el tráfico entrante a *Niles* y los correspondientes tiempos de respuesta, mientras que en la parte inferior se ilustran los mismos parámetros para *Favourites-service*. Las peticiones de *Niles* hacia *Favourites-service* son continuas, pero como éste último degrada sus tiempos de respuesta, el menú sufre una penalización. Una vez que se incorpora el *Circuit Breaker*, la Fig. 7 muestra la reducción de las peticiones hacia *Favourites-service* durante su degradación, lo que conduce al ahorro de recursos. Además, en la Fig. 8 se puede observar la disminución de los tiempos de respuesta del menú durante el lapso que no se envían peticiones a *Favourites-service*. Esta mejora se debe a que no se continúa esperando por una respuesta que probablemente no pueda resolverse.

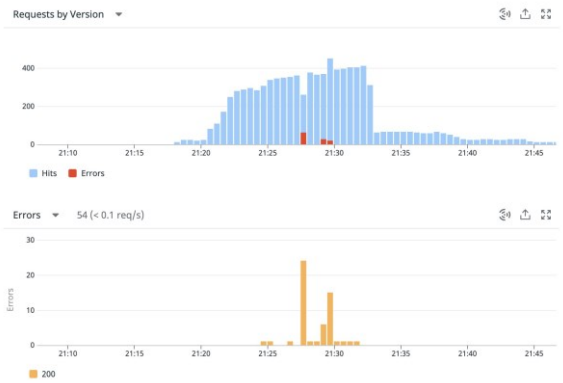


Figura 5: Peticiones hacia *Niles* (arriba), y respuestas código de estado 200 (abajo), con *Retry*

Los rectángulos marcados en la Fig. 8 ilustran los beneficios de la aplicación de *Circuit Breaker*. El rectángulo de la izquierda destaca el lapso de tiempo en el que no se realizan peticiones hacia *Favourites-service*, debido a que el *Circuit Breaker* pasa a estado abierto. El rectángulo del centro permite ver que los tiempos de respuesta de *Favourites-service* durante el incidente superan los 5000 ms; mientras que el rectángulo de la derecha evidencia que el tiempo de respuesta del menú no supera los 500 ms. Esto representa una mejora de 10 veces respecto a no aplicar el disyuntor. Por lo tanto, la aplicación del patrón permite, además del ahorro de recursos, obtener mejoras en el desempeño del servicio si, por ejemplo, se omite la petición para obtener los productos favoritos y se retorna un menú carente de esa información, debido a que el interruptor está abierto.

5. Conclusiones y Trabajo Futuro

En la actualidad, la resiliencia es uno de los aspectos no funcionales más buscados en los sistemas considerando el impacto directo con el negocio de una empresa. En este trabajo se analiza la aplicación de una serie de patrones que se utilizan en la empresa



Figura 6: Tráfico entrante y tiempos de respuesta en *Niles* (arriba) y *Favouritesservice* (abajo)

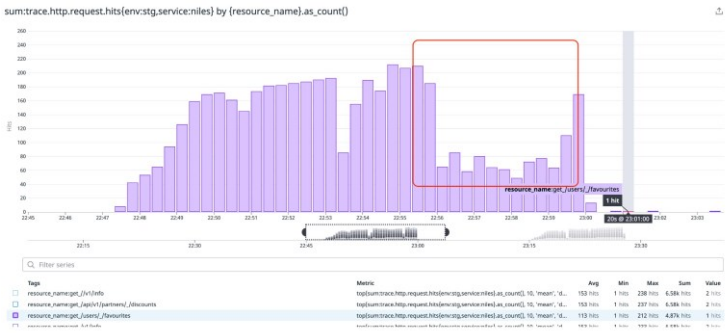


Figura7: Ahorro de recursos con la ayuda del patrón *Circuit Breaker*

PedidosYa para afrontar diversos fallos capaces de afectar el funcionamiento de su ecosistema de microservicios. A partir de los resultados experimentales, se puede concluir que:

- *Timeout* permitió que el tiempo de respuesta para retornar el menú se mantenga razonable, ya que con la inclusión del patrón se disminuyó de aproximadamente 6000 a 170ms (aprox. 35×). Además, al finalizar la comunicación en tiempo acotado, se evita la acumulación de requerimientos pendientes, mejorando así el uso de recursos.
- *Retry* permitió que una mayor cantidad de las peticiones entrantes puedan ser resueltas exitosamente, reduciendo la cantidad de requerimientos erróneos de 822 a 54ms (aproximadamente 15×). *Retry* muestra que, en algunos contextos, un simple reintento puede completar exitosamente una funcionalidad que impacta en las ganancias de la compañía (como la facturación de una orden).

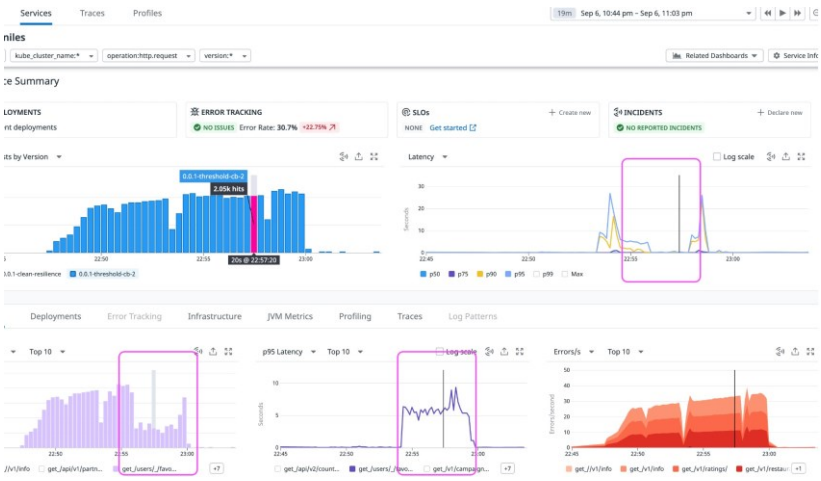


Figura8: Mejoras al aplicar el patrón *Circuit Breaker*

- *Circuit Breaker* facilitó la provisión de un menú disminuido (sin favoritos) ante situaciones de elevadas probabilidades de falla. Además de permitir un ahorro de recursos, evitó realizar peticiones hacia aquellos servicios temporalmente averiados e incluso redujo el tiempo de respuesta del menú en ciertos escenarios.

La implementación de cada uno de estos patrones de resiliencia en *Niles* se encuentran en producción en el ecosistema de PedidosYa, constituyéndolo como uno de los componentes más utilizados y robustos dentro de la compañía. Como trabajo futuro, se propone extender el análisis de la implementación de otros patrones orientados a la resiliencia como *Bulkhead*, *Throttling* o *Queue-Based Load Leveling*, además de considerar la combinación de los ya contemplados.

Referencias

1. Fowler, M.: Microservices, https://martinfowler.com/articles/microservices. html
2. Hofmann, M., Schnabel, E., Stanley, K., Organization, I.B.M.C.I.T.S.: Microservices Best Practices for Java. IBM redbooks, IBM Corporation, International Technical Support Organization (2016), https://books.google.com.ar/books?id=ZiSJAQAACAAJ
3. ION Group: How Do You Manage Non-Functional Requirements?, https://iongroup.com/blog/markets/how-do-you-manage-non-functional-requirements/
4. Nygard, M.T.: Release it! design and deploy production-ready software, Second edition. Pragmatic Bookshelf (2018)
5. Peter Jausovec: Fallacies of distributed systems, https://blogs.oracle.com/developers/post/fallacies-of-distributed-systems
6. Richardson, C.: Microservices Patterns: With examples in Java, chap. Escaping monolithic hell. Manning (2018), https://books.google.com.ar/books?id=UeK1swEACAAJ
7. Richardson, C.: Microservices Pattern: Circuit Breaker, https://microservices.io/patterns/reliability/circuit-breaker.html
8. Suárez, S.L.: Análisis de patrones de resiliencia en una arquitectura basada en microservicios. Tesina de Licenciatura en Sistemas, Universidad Nacional de La Plata (Feb 2023), http://sedici.unlp.edu.ar/handle/10915/149187

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Adaptación de Algoritmo OpenMP para Computar Caminos Mínimos en Grafos en Arquitecturas x86

Sergio Calderón[1] [0000-0001-6736-7358], Enzo Rucci[1], and Franco Chichizola[1][0000-0001-8857-6343]

III-LIDI, Facultad de Informática, UNLP – CIC.
La Plata (1900), Bs As, Argentina
{scalderon,erucci,francoch}@lidi.info.unlp.edu.ar

Resumen Los grafos han adquirido una relevancia significativa para modelar y resolver problemas en diversas áreas. El algoritmo FloydWarshall (FW) permite hallar los caminos mínimos entre vértices. Es una solución de alta demanda computacional ($O(n^3)$), debiendo emplear cómputo paralelo cuando el tamaño del problema escala. En este trabajo, se presenta la optimización de FW en arquitecturas multicore x86 de propósito general, adaptando un código diseñado para un acelerador específico (Xeon Phi KNL). Se parte desde una versión paralela que emplea una técnica de blocking, y luego se describen las mejoras incrementales aplicadas. Las pruebas realizadas en un servidor con 2×Intel Xeon Platinum 8276L y en un equipo comercial con Intel Core i5-10400F muestran mejoras acumuladas de 7.31× y 6.98×, respectivamente. Todas las optimizaciones resultan beneficiosas, aunque con distinto impacto. Por último, se plantea la idea de una nueva optimización FW.

Keywords: Floyd-Warshall · Multicore · HPC · Caminos mínimos · Xeon · Xeon Phi Knights Landing · Core · OpenMP

1. Introducción

El algoritmo de Floyd-Warshall (FW) [2,16] permite calcular y hallar los caminos mínimos entre todos los vértices de un grafo pesado. Es por lo que se lo ha empleado en ámbitos diversos como el tráfico automovilístico [6], las redes de computadoras [7], bioinformática [8], computación gráfica [15], entre otros. Sin embargo, FW es computacionalmente costoso ($O(n^3)$) y a medida que el tamaño del problema escala, el empleo de recursos de cómputo paralelo se vuelve necesario para poder satisfacer sus requerimientos. Es por lo que la comunidad científica ha realizado múltiples esfuerzos con ese propósito [9,14,4,3,17,13,12]. En particular para las arquitecturas Xeon Phi de Intel, Rucci *et al.* [11] exploró su uso para acelerar FW en la primera generación (KNC, Knights Corner), mientras que Costi *et al.* [1] lo extendió a la segunda (Knights Landing, KNL).

En este artículo, se propone tomar el trabajo realizado por Costi *et al.* [1] y adaptar su código para que pueda ejecutarse en procesadores multicore Intel x86, abandonando la especificidad del Xeon Phi KNL. Para ello, se verificará una a una las optimizaciones propuestas por [1], realizando ajustes al código de base donde sea necesario, y

analizando su rendimiento en dos servidores de referencia ante diferentes escenarios de prueba. Resulta importante mencionar que este trabajo representa el primer paso hacia la propuesta de nuevas optimizaciones para FW y el posterior desarrollo de una librería de código especializada.

El resto del artículo se organiza de la siguiente forma. La Sección 2 introduce el marco referencial para este trabajo. Luego, la Sección 3 describe la implementación realizada. A continuación, la Sección 4 muestra los resultados experimentales obtenidos. Finalmente, la Sección 5 enumera las líneas a futuro y la Sección 6 resume las conclusiones del trabajo.

2. Marco referencial

2.1 Intel Xeon Phi

Xeon Phi es el nombre comercial que Intel usó para una serie de procesadores many-core orientados a HPC. En 2012, Intel lanzó la primera generación (KNC) que principalmente contaba con hasta 61 núcleos Pentium x86 con unidades vectoriales extendidas (512 bits, específicas para KNC) e Hyper-Threading (cuatro hilos de hardware por núcleo). Mientras KNC se conectaba al procesador host a través del bus PCI Express, la segunda generación (KNL) pudo funcionar como procesador independiente. Entre sus principales características, se pueden mencionar la mayor cantidad de núcleos con soporte para hyper-threading (hasta 72), la incorporación de las instrucciones vectoriales AVX-512 (también de 512 bit pero compatibles con procesadores Xeon) y la integración de una memoria de alto ancho de banda, denominada MCDRAM [10]. La última generación (Knights Mill, KNM) fue lanzada a finales de 2017, siendo una variante de KNL con instrucciones específicas para aprendizaje automático profundo. Finalmente, Intel anunció que discontinuaría la serie Xeon Phi en 2018 para dedicarse al desarrollo de placas gráficas (GPUs)[1].

2.2 Intel Xeon y Core

En la actualidad, Intel presenta dos gamas de procesadores de la familia x86: Xeon y Core. Los procesadores Intel Xeon están diseñados para tareas empresariales y de servidor que requieren alta potencia de procesamiento y confiabilidad, mientras que los procesadores Intel Core son ideales para uso general, incluidos juegos, aplicaciones ofimáticas y entretenimiento multimedia.

En cuanto a las características arquitectónicas, los Xeon suelen tener más núcleos, tecnologías específicas para virtualización y seguridad, soporte para configuración en múltiples sockets, entre otras características avanzadas. En sentido contrario, los procesadores Core suelen tener frecuencias más altas, menor consumo energético y menor precio.

[1] https://hardzone.es/2018/07/25/intel-adios-xeon-phi-reemplazados-tarjetas-graficas/

La elección entre Xeon y Core dependerá de las necesidades específicas de la aplicación, el presupuesto y las preferencias del usuario.

2.3 Algoritmo FW

El pseudo-código de FW se muestra en la Fig. 1. Dado un grafo G de N vértices, FW recibe como entrada una matriz densa D de $N \times N$ que contiene las distancias entre todos los pares de vértices G, donde $D_{i,j}$ representa la distancia del nodo i al nodo j[2]. FW computa N iteraciones, evaluando en la k-ésima iteración todos los posibles caminos entre los vértices i y j que tienen a k como vértice intermedio. Como resultado, produce una matriz actualizada D, donde $D_{i,j}$ contiene ahora la distancia mínima entre los nodos i y j hasta ese paso. Complementariamente, FW va construyendo una matriz adicional P que registra los caminos asociados a las distancias mínimas.

Algoritmo FW por bloques. A primera vista, la estructura de triple bucle anidado de este algoritmo resulta similar al de la multiplicación de matrices densas (MM). Sin embargo, como las lecturas y escrituras se realizan sobre la misma matriz, los tres bucles no pueden ser intercambiados libremente, como es el caso de MM. A pesar de esto, el algoritmo FW puede ser computado por bloques bajo ciertas condiciones [14].

```
for k ← 0 to N − 1 do
  for i ← 0 to N − 1 do
    for j ← 0 to N − 1 do
      if D_{i,j} > D_{i,k} + D_{k,j} then
        D_{i,j} ← D_{i,k} + D_{k,j}
        P_{i,j} ← k
      end if
    end for
  end for
end for
```

Figura 1: Pseudocódigo del algoritmo de FW

El algoritmo FW por bloques (FWB) divide a la matriz D en bloques de tamaño $TB \times TB$, totalizando $(N/TB)^2$ bloques. El cómputo se organiza en $R = N/TB$ rondas, donde cada ronda consta de 4 fases ordenadas de acuerdo a las dependencias de datos entre los bloques:

- Fase 1: actualizar el bloque $D^{k,k}$ debido a que sólo depende de sí mismo.
- Fase 2: actualizar los bloques de la fila k de bloques ($D^{k,*}$) debido a que cada uno de estos depende de sí mismo y de $D^{k,k}$.
- Fase 3: actualizar los bloques de la columna k de bloques ($D^{*,k}$) debido a que cada uno de estos depende de sí mismo y de $D^{k,k}$.
- Fase 4: actualizar los bloques restantes $D^{i,j}$ de la matriz porque cada uno depende de los bloques $D^{i,k}$ y $D^{k,j}$ de su fila y columna de bloques, respectivamente.

En la Fig. 2 se muestra gráficamente cada una de las fases de cómputo y las dependencias entre bloques. Los bloques en amarillo son aquellos bloques que están siendo computados, los grises los que ya se procesaron, y los verdes representan los que faltan computar.

[2] Si no existe un camino entre los nodos i y j, se asigna *infinito* a su distancia (usualmente representado como el valor positivo más grande)

2.4 Código de base

Como código de base se empleó el de [1], el cual fue desarrollado específicamente para procesadores de la arquitectura Xeon Phi KNL de Intel [10]. Para la implementación final del algoritmo FW con blocking en esta arquitectura paralela se realizaron diversos análisis. Las mejoras encontradas se fueron combinando a modo de incrementos con la nomenclatura *Opt-X*, los cuales se describen a continuación.

- **Opt-0: Granularidad utilizada**. Se elige el `loop` a paralelizar, dando lugar a una paralelización intra-bloque (de grano fino) o inter-bloque (de grano grueso). La primera se implementa en la fase 1, ya que computa un único bloque. En el resto de fases se usa la forma inter-bloque, ejecutando en paralelo las fases 2 y 3 por ser independientes entre sí, y luego la fase 4.
- **Opt-1: MCDRAM**. Es una memoria adicional presente en el procesador Xeon Phi KNL, caracterizada por brindar un ancho de banda alto (hasta 450 GB/s). Sin embargo, tiene poca capacidad, por lo que se usa el comando numactl -p para asignar de forma auxiliar en la DDR.
- **Opt-2: Vectorización guiada por SSE**. Mediante la directiva `simd` de OpenMP, se indica al compilador que fuerce la vectorización de un bucle. Habitualmente, los compiladores suelen utilizar por defecto el conjunto de instrucciones SSE de 128 bits, lo que permite procesar hasta 2 datos de tipo *double* o 4 de tipo *float*.
- **Opt-3: Vectorización guiada por AVX2**. Usando el mismo código, se le indica al compilador el parámetro `-xAVX2` para que se utilicen instrucciones AVX de 256 bits, lo que permite duplicar la cantidad de operaciones simultáneas respecto a SSE (si está soportado).

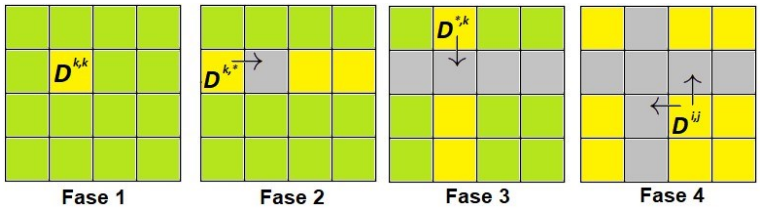


Figura2: Fases del cómputo de FWB y sus dependencias

- **Opt-4: Vectorización guiada por AVX-512**. ídem anterior, pero el flag indicado es `-xMIC-AVX512`, para emplear instrucciones de 512 bits. De esta forma, se pueden procesar hasta 8 datos tipo *double* o 16 tipo `float` en simultáneo en un procesador que lo soporte.
- **Opt-5: Alineación de datos**. Mediante la función `_mm_malloc()` de Intel se reserva memoria de modo que los datos se almacenen alineados respecto al comienzo de cada línea de caché. Indicando el ancho de línea de la cache L1D al reservar memoria, se optimizan las lecturas y escrituras posteriores.
- **Opt-6: Predicción de saltos**. La función `_builtin_expect(exp, c)` indica al compilador el resultado más probable (*c*) en una sentencia IF cuya expresión es (*exp*). La comparación entre distancias es un hotspot clave de FW, que en la

mayoría de veces será *false* (no se actualiza *D*). Mientras más acierte el planificador, más paralelismo a nivel de instrucciones podrá aprovechar el procesador.

- **Opt-7: Desenrollado guiado de bucles**. Realizado mediante la directiva `unroll(FD)`, se reduce la cantidad de iteraciones de un bucle según un factor convenientemente elegido para explotar la vectorización disponible.
- **Opt-8: Afinidad de hilos**. Se indica cómo realizar la distribución de hilos entre los núcleos según la variable de entorno *KMP_AFFINITY*. Se especifica un tipo de afinidad (`balanced`, `compact` o `scatter`) y la granularidad (*fine*, *core* o *tile*). En Xeon Phi KNL, la configuración óptima fue *fine, balanced*.

3. Implementación

El código de base explicado en la Sección 2 fue adaptado para su ejecución en los siguientes equipos de distintas prestaciones:

- Un equipo de gama comercial con procesador Intel Core i5-10400F de arquitectura x86. Posee 6 núcleos (2 hilos hw por núcleo) de frecuencia base de 2.90 GHz y 32 GB de memoria RAM. El sistema operativo es Debian GNU/Linux 11.
- Un equipo de alto rendimiento constituido por dos procesadores Intel Xeon Platinum 8276L en Dual-Socket. Cada procesador posee 28 cores de 2.20 GHz, nuevamente con 2 hilos hardware cada uno, resultando un total de 112 hilos. Dispone de un total de 250 GB de memoria RAM y la distribución de Linux instalada es Ubuntu 20.04 LTS.

En ambos casos, se empleó el compilador Intel ICC, parte de la suite oneAPI versión 2021.7.1. A continuación, se detallan las modificaciones realizadas a las optimizaciones para su correcta adaptación en cada equipo.

- **Opt-0:** se mantiene el tipo de paralelización original, por tanto, las directivas *for* de OpenMP y el orden de las fases permanecen sin cambios.
- **Opt-1:** descartado en ambos casos al no contar con memoria MCDRAM, por lo que se omite el comando `numactl -p` en las pruebas.
- **Opt-2 y Opt-3:** sin cambios para las vectorizaciones por SSE y AVX2.
- **Opt-4:** en Intel Xeon Platinum se reemplaza el flag asociado por el recomendado a dicho procesador (de `-xMIC-AVX512` a `-xCORE-AVX512`); mientras que en Intel Core i5 se descarta por no contar con este nivel de vectorización.
- **Opt-5:** en Intel Xeon Platinum se mantiene el parámetro SIMD WIDTH en el valor 512; mientras que en Intel Core i5 se reduce a 256 por ser AVX-2 la máxima vectorización disponible, además de usar el flag de Opt-3.
- **Opt-6:** sin cambios en las macros *likely(exp)* y *unlikely(exp)*.
- **Opt-7:** el factor *FD* depende de SIMD WIDTH y TYPE SIZE, parámetros definidos en compilación. El valor de este último es 32 para simple precisión (*float*) y 64 para doble precisión (*double*).
- **Opt-8:** se elige la afinidad y granularidad óptima de manera empírica.

4. Trabajo experimental y resultados obtenidos

4.1 Pruebas realizadas

Las pruebas se realizaron sobre los equipos descritos en la Sección 3 una vez aplicadas las modificaciones. Para ambos equipos se consideró la variación del tamaño de la matriz de distancias (N = {4096, 8192, 16384}), el tamaño de bloque (TB = {32, 64, 128, 256}) y el tipo de dato (*float, double*). Además, se emplearon diferente cantidad de hilos OpenMP, considerando dos estrategias: un hilo por núcleo físico y uno por núcleo lógico (T_{i5} = {6,12}, T_{Xeon}={56,112}).

El programa inicia con la carga del grafo para la creación de las matrices D y P. Se utiliza la misma entrada para todas las pruebas que se realicen con un determinado N y TB. Se estableció un porcentaje de completitud del 70% para todos los casos. Cada prueba particular fue repetida 8 veces, calculando los promedios para reducir la variabilidad.

4.2 Resultados experimentales

Como métrica de rendimiento, se emplea $GFLOPS = \frac{2 \times N^3}{t \times 10^9}$, donde N es la cantidad de vértices del grafo, t es el tiempo de ejecución (en segundos), y 2 representa la cantidad de operaciones en punto flotante que se ejecutan en cada iteración del bucle más interno del algoritmo.

Considerando un tamaño de entrada intermedio (N = 8192) y tipo *float*, se presentan los resultados obtenidos para cada equipo en las Fig. 3 (Core i5) y 4 (Xeon). Las columnas indican la versión utilizada, y en las filas se agrupa por número de hilos T y luego se indica el tamaño de bloque TB. Se puede observar que el mejor rendimiento para ambos equipos se logra usando 1 hilo por núcleo lógico y TB =128 desde Opt-3 en adelante.

Promedio de GFlops	opt_0_1	opt_2	opt_3	opt_5	opt_6	opt_7	opt_8
6							
32	12,35	20,62	61,92	65,95	78,89	103,11	103,44
64	14,37	19,38	73,58	78,41	95,47	136,61	137,42
128	19,99	23,40	77,06	83,50	104,07	146,66	146,70
256	20,34	24,61	55,09	56,99	86,83	110,86	111,34
12							
32	16,24	28,15	66,85	70,26	87,15	112,24	112,65
64	19,99	25,52	78,29	81,36	101,56	146,24	147,05
128	21,09	29,92	78,41	82,82	107,65	154,22	154,29
256	21,22	32,13	60,51	62,12	94,19	110,68	110,80

Figura 3: GFLOPS promedio en Core i5 para N=8192 y tipo *float*

Promedio de GFlops	opt_0_1	opt_2	opt_3	opt_4	opt_5	opt_6	opt_7	opt_8
56								
32	62,03	111,06	116,07	185,12	222,85	346,40	374,78	494,90
64	62,95	81,35	111,03	189,76	230,82	472,82	480,55	710,57
128	87,15	110,83	132,55	215,53	334,13	507,52	559,67	831,70
256	87,90	111,65	112,06	185,94	232,88	441,03	444,93	641,24
112								
32	79,23	157,39	224,92	261,30	273,38	422,60	440,06	463,17
64	78,10	101,26	402,67	447,31	491,73	587,05	611,58	664,20
128	124,09	163,40	425,86	489,67	593,25	700,64	766,82	866,31
256	124,47	169,32	336,85	354,80	372,23	432,98	456,85	470,91

Figura 4: GFLOPS promedio en Xeon Platinum para N=8192 y tipo *float*

En la Tabla 1 se especifica la mejora obtenida en cada *Opt-X* respecto a su anterior, incluyendo también la comparación con [1]. Las Fig. 5 y 6 muestran el rendimiento alcanzado utilizando la configuración óptima mencionado en el equipo Core i5 y el Xeon, respectivamente. Se puede observar en estas figuras que efectivamente hay un incremento de los GFLOPS obtenidos al avanzar de optimización en ambos equipos. Se observa que el mayor salto de aceleración se encuentra en la versión *Opt-3* mediante la vectorización con *AVX-2*, siendo de **2.6x** aproximadamente. Luego, al considerar la opción de mayor vectorización frente a la que no aprovecha esta optimización (*Opt-0*), se obtiene una mejora total de **3.96x** en Xeon Platinum y **3.72x** en Core i5 (se debe tener en cuenta que este último no admite la vectorización *AVX-512*).

	Opt-1	Opt-2	Opt-3	Opt-4	Opt-5	Opt-6	Opt-7	Opt-8
Core i5	-	1.42x	2.62x	-	1.06x	1.30x	1.43x	1.0005x
Xeon Platinum	-	1.32x	2.61x	1.15x	1.21x	1.18x	1.09x	1.13x
Xeon Phi KNL	1.03x	1.57x	2.19x	2.10x	1.05x	2.63x	1.40x	1.0012x

Tabla 1: Mejora incremental en cada equipo para N=8192

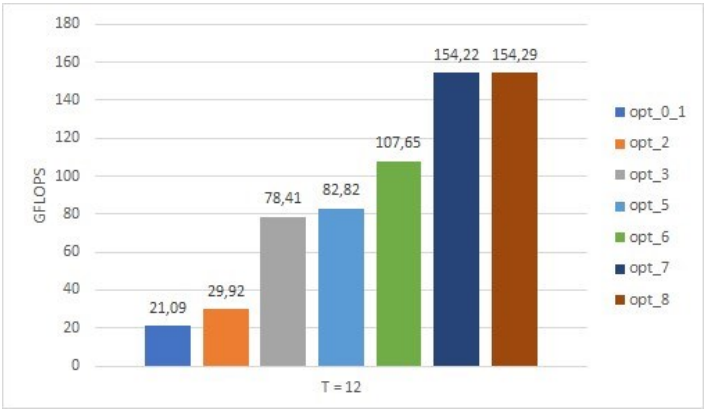


Figura 5: GFLOPS en Core i5 para tipo *float*, configuración óptima

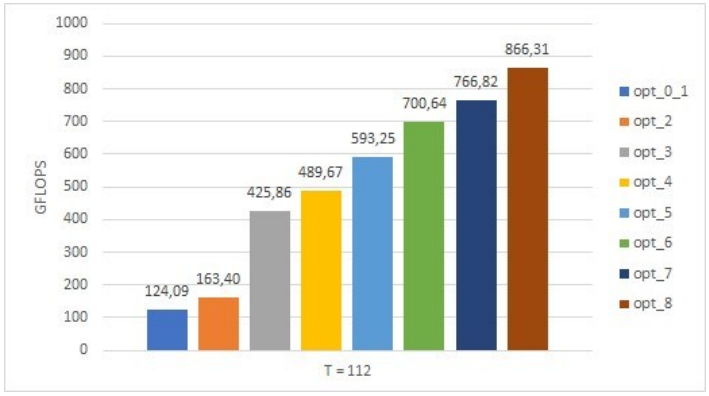


Figura 6: GFLOPS en Xeon Platinum para tipo *float*, configuración óptima

La siguiente mejora destacable ocurre en la predicción de saltos de *Opt-6*, alcanzándose un **1.30x** para Core i5, y **1.18x** para Xeon Platinum. Su optimización posterior *Opt-7*, que consiste en el desenrollado de bucles, brinda una aceleración similar, siendo **1.43x** y **1.09x** en los equipos mencionados. Para la versión *Opt-8* se probaron las 6 combinaciones de granularidad (core, fine) y afinidad (balanced, compact y scatter) [5]. Se encontró que la mejor configuración es **fine-balanced**, por leve diferencia sobre *scatter*, como se observa en la Fig. 7 usando la mitad de hilos disponibles y tipo float. Cuando se dispone de pocos núcleos, en la última optimización se puede medir una diferencia de GFLOPS mínima en comparación a *Opt-7*, lo cual se ve reflejado en el equipo comercial con una mejora de **1.0005x**.

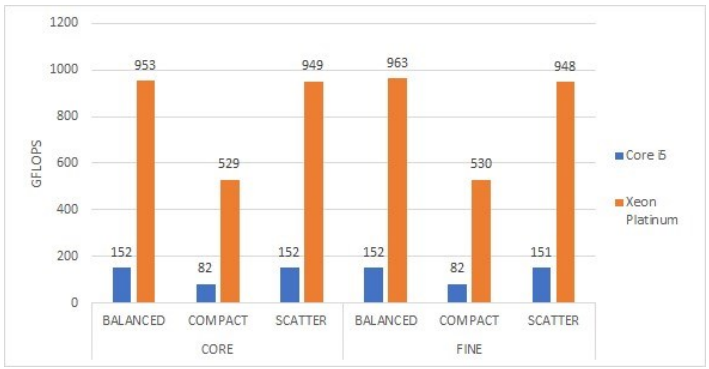


Figura 7: GFLOPS para N=16384 según KMP AFFINITY

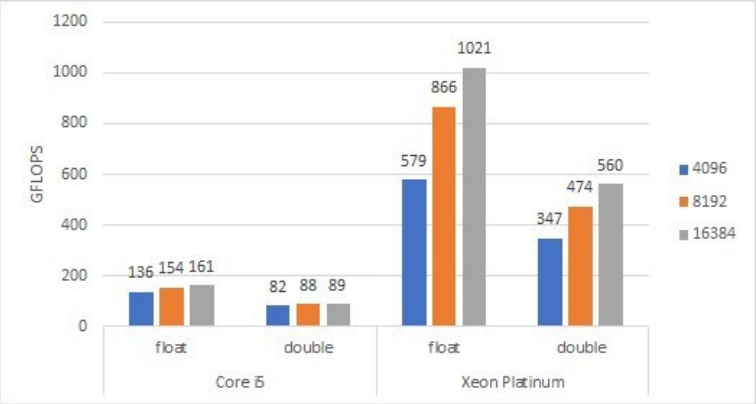


Figura 8: GFLOPS en Opt-8 para tipo float y double

En resumen, si se considera el desempeño integral de las versiones tomando como base a *Opt-0*, al sumar las mejoras obtenidas desde *Opt-2* hasta *Opt8*, se tiene un acumulado de **7.31x** y **6.98x** para Core i5 y Xeon Platinum, respectivamente. Adicionalmente, en la comparación con el Xeon Phi KNL, se puede notar que todas las optimizaciones resultaron beneficiosas, aunque no todas impactaron de la misma manera.

Por último, en la Fig. 8 se muestran los GFLOPS alcanzados con la mejor versión para cada equipo, tipo de dato y tamaño de entrada testeado, empleando la configuración óptima de *T* y *TB* a cada caso. En primer lugar, resulta claro que se obtienen mayores GFLOPS en el equipo Xeon que en el Core i5, considerando su potencia de cómputo. En segundo lugar, se observa que el rendimiento se incrementa a medida que aumenta *N*, dado a la mayor proporción de cómputo frente a sincronización. En particular, la magnitud de esta diferencia es más notoria cuando se dispone de más hilos (caso Xeon Platinum). En tercer lugar, el uso de un tipo de dato de mayor precisión como *double* puede llevar a un resultado más fiable; sin embargo, se debe tener en cuenta que tendrá un costo en el tiempo de respuesta, ya que el rendimiento decae hasta un 45%.

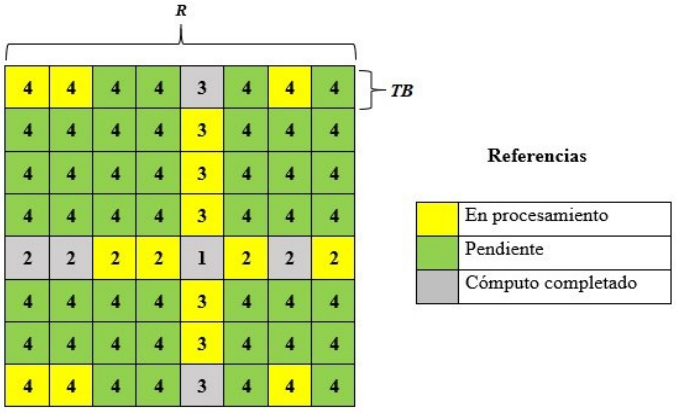


Figura9: Ejemplo de oportunidad de optimización para $R = 8$ en la ronda $k = 4$

5. Trabajo futuro

A partir del trabajo desarrollado y de los resultados encontrados, se mencionan algunos de los próximos pasos a seguir:

- Desarrollar nuevas versiones incrementales y realizar las pruebas correspondientes para determinar si se pueden obtener nuevas mejoras. En el algoritmo FWB presentado, se da comienzo a la fase 4 de una ronda sólo una vez que fueron completadas las fases 2 y 3. Una idea propuesta es reducir esta espera ociosa ya que se requiere satisfacer únicamente las dependencias directas de cada bloque de fase 4 en particular, mientras se continúa el cómputo de las fases 2 y 3. La Fig. 9 ejemplifica la oportunidad de optimización, donde los bloques ya computados se muestran en color gris (5 de la fase 2-3), y aquellos en procesamiento en amarillo (6 de la fase 4). Esta posible optimización toma mayor relevancia cuando *T* es *grande* y *TB* es *chico*, y su implementación probablemente requiera una sincronización de grano más fino que la que las directivas de OpenMP pueden proveer, siendo necesario trabajar a nivel de Pthreads.
- Realizar modificaciones al código con el fin de posibilitar su compilación mediante el nuevo compilador **ICX** de Intel, el cual incorpora a LLVM [3]como *backend*, manteniendo la filosofía de las optimizaciones incrementales (una versión siempre agrega una mejora respecto a su predecesor).
- Desarrollar una librería para facilitar la inclusión y utilización del algoritmo paralelo FW en programas de C/C++, con sus optimizaciones internamente implementadas, que permita especificar los parámetros de aplicación que resulten necesarios.

6. Conclusiones

En este artículo, se tomó el trabajo realizado por Costi*et al.* [1] y se adaptó su código para que pueda ejecutarse en procesadores multicore Intel x86, abandonando la especificidad del Xeon Phi KNL. Para ello, se verificó una a una las optimizaciones propuestas por Costi *et al.* [1], realizando ajustes al código de base donde fue necesario, y se analizó su rendimiento en dos servidores de referencia ante diferentes escenarios de prueba. A partir de los resultados obtenidos y su posterior análisis, se pueden mencionar las siguientes conclusiones:

- Entre las adaptaciones realizadas para Xeon Platinum se encuentra la actualización de un *flag* de compilación, precisamente el utilizado para la vectorización con AVX-512 en la versión *Opt-4*; y la omisión de *Opt-1* debido a la inexistencia de la memoria adicional MCDRAM.
- En el equipo con Intel Core i5, se debió suprimir la optimización *Opt-4* además de la mencionada *Opt-1*, ya que la máxima vectorización disponible es AVX2.

[3] The LLVM Compiler Infrastructure Project https://llvm.org/

- Todas las optimizaciones aplicadas originalmente al Xeon Phi KNL también resultaron beneficiosas en los dos equipos x86 testeados. Las mayores ganancias se obtienen en las optimizaciones que añaden la vectorización de instrucciones.

Habiendo realizado la adaptación del código original, se ha planteado la idea de una nueva posible optimización para FW y su compatibilización con el nuevo compilador de Intel. Posteriormente, se planea el desarrollo de una librería que facilite el uso de estos algoritmos en aplicaciones externas. Finalmente, vale la pena destacar que el código fuente se encuentra público en un repositorio de GitHub, accesible mediante el siguiente link: https://bit.ly/cacic-2023-fw

Referencias

1. Costi, U.: Aceleración del Algoritmo Floyd-Warshall sobre Intel Xeon Phi KNL. Tesina de Licenciatura en Informática, Universidad Nacional de La Plata (2020)
2. Floyd, R.W.: Algorithm 97: Shortest path. Commun. ACM **5**(6), 345– (Jun 1962). https://doi.org/10.1145/367766.368168
3. Han, S.C., Franchetti, F., Püschel, M.: Program generation for the all-pairs shortest path problem. In: Proceedings of the 15th International Conference on Parallel Architectures and Compilation Techniques. p. 222–232. PACT '06, ACM, New
4. York, NY, USA (2006). https://doi.org/10.1145/1152154.1152189
5. Han, S., Kang, S.: Optimizing all-pairs shortest-path algorithm using vector instructions (2006)
6. Intel Corporation: Thread Affinity Interface, https://www.intel.com/content/www/us/en/develop/documentation/cpp-compiler-developer-guide-and-reference/top/optimization-and-programming/openmp-support/openmp-library-support/ thread-affinity-interface.html
7. Jalali, S., Noroozi, M.: Determination of the optimal escape routes of underground mine networks in emergency cases. Safety Science **47**(8), 1077 – 1082 (2009).
8. https://doi.org/http://dx.doi.org/10.1016/j.ssci.2009.01.001
9. Khan, P., Konar, G., Chakraborty, N.: Modification of floyd-warshall's algorithm for shortest path routing in wireless sensor networks. In: 2014 Annual IEEE India Conference (INDICON). pp. 1–6 (Dec 2014). https://doi.org/10.1109/INDICON.2014.7030504
10. Nakaya, A., Goto, S., Kanehisa, M.: Extraction of correlated gene clusters by multiple graph comparison. Genome Informatics **12**, 44–53 (2001)
11. Penner, M., Prasanna, V.K.: Cache-friendly implementations of transitive closure. In: Proceedings of the 2001 International Conference on Parallel Architectures and Compilation Techniques. pp. 185–. PACT '01, IEEE Computer Society, Washington, DC, USA (2001)
12. Reinders, J., Jeffers, J., Sodani, A.: Intel Xeon Phi Processor High Performance Programming Knights Landing Edition. Morgan Kaufmann Publishers Inc., Boston, MA, USA (2016)
13. Rucci, E., De Giusti, A., Naiouf, M.: Blocked All-Pairs Shortest Paths Algorithm on Intel Xeon Phi KNL Processor: A Case Study. In: De Giusti, A.E. (ed.) Computer Science – CACIC 2017. pp. 47–57. Springer Int. Pub., Cham (2018)
14. Solomonik, E., Buluc¸, A., Demmel, J.: Minimizing communication in all-pairs shortest paths. In: 2013 IEEE 27th International Symposium on Parallel and Distributed Processing. pp. 548–559. IEEE (2013)
15. Srinivasan, T., Balakrishnan, R., Gangadharan, S., Hayawardh, V.: A scalable parallelization of all-pairs shortest path algorithm for a high performance cluster

environment. In: 2007 International Conference on Parallel and Distributed Systems. pp. 1–8 (2007). https://doi.org/10.1109/ICPADS.2007.4447721

16. Venkataraman, G., Sahni, S., Mukhopadhyaya, S.: A Blocked All-Pairs Shortest-Paths Algorithm, pp. 419–432. Springer Berlin Heidelberg (2000). https://doi.org/10.1007/3-540-44985-X 36
17. Wang, L., Springer, M., Heibel, H., Navab, N.: Floyd-warshall all-pair shortest path for accurate multi-marker calibration. In: 2010 IEEE International Symposium on Mixed and Augmented Reality. pp. 277–278 (2010). https://doi.org/10.1109/ISMAR.2010.5643605
18. Warshall, S.: A theorem on boolean matrices. J. ACM **9** (1), 11–12 (Jan 1962). https://doi.org/10.1145/321105.321107
19. Zhang, L.y., Jian, M., Li, K.p.: A parallel floyd-warshall algorithm based on tbb. In: 2010 2nd IEEE International Conference on Information Management and Engineering. pp. 429–433 (2010). https://doi.org/10.1109/ICIME.2010.5477752

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Evaluation of FaaS as an alternative to build HPC environments

María Murazzo, Joaquín Lebetti, Nelson Rodríguez, Adriana Martin

Departamento e Instituto de Informática - F.C.E.F. y N. - U.N.S.J.
Complejo Islas Malvinas. Cereceto y Meglioli. 5400. Rivadavia. San Juan,
lebejoaquin@gmail.com, marite@unsj-cuim.edu.ar,
nelson@iinfo.unsj.edu.ar , adrianamartin1@gmail.com

Abstract. Running HPC applications in the cloud has proven to be a viable option to conventional parallel or distributed architectures, which require a high degree of management as well as poor resource scaling. The traditional approach for a user is to usually use the Cloud provider to provision resources to virtual machines (VMs), using them in a similar way to an on-premises infrastructure, with the consequent problem of resource management coupled with the degradation of application performance due to contextualization of virtualized environments. Serverless computing allows a user to run code written in the programming language of their choice, without first having to provision a virtual machine. On the other hand, elasticity, availability, scalability, and fault tolerance are provided transparently by the cloud provider. This way it is possible to reduce the complexity of infrastructure management for the developer, allowing him to focus on the logic of the application. Also, economic advantages arise, when just paying for usage time. The work focuses on the challenge of evaluating the cost, not only monetary but also of performance, of migrating HPC applications to serverless environments. This evaluation will allow the decision to be made REGARDING which infrastructure will be used, in order to obtain the best performance benefit.

Keywords: IoT, Cloud Computing, HPC Serverles Computing, FaaS, GCP, Cloud Function.

1. Introduction

The increasing popularity of IoT and the massification of cloud infrastructures have recently opened up a world of possibilities for HPC applications. This is due to the huge quantity of data that IoT devices generate, which makes it impractical to treat them with traditional paradigms. To achieve the adequate processing of this kind of data with significant speed and size characteristics, it is mandatory to dispense with traditional programming paradigms [1]. Therefore, it is necessary to apply algorithms to take advantage of the scalability of computing and data processing resources [2] [3]. In this sense, it is proposed high performance computing techniques (HPC) as a solution to data processing coming from IoT so as to increase processing performance.

Although HPC architectures have evolved to get better response times for applications, they have the drawback of scaling computing resources, which is why migrating to the cloud becomes a reasonable alternative [4].

Like any other service, Cloud Computing has been characterized as a technology focused on providing on-demand computing, which is an advantage for assembling applications where intensive processing is necessary [5].

Cloud providers claim many advantages in HPC application migration, such as fast access to resources, lower costs and flexibility in sourcing and provisioning [6]. However, the cloud has two major disadvantages, the first one being the degradation of the applications performance when assembled on virtualized architecture, since it generates overhead in the contextualization of virtual machines; and the second disadvantage when deploying applications in the cloud, is that the organization must be responsible for keeping the infrastructure needed to deploy the applications working properly, which leads to charging costs on the budget for its maintenance and support [7].

In this sense, the emergence of Serverless Computing [8] means that developers do not have to worry about infrastructure provisioning and scaling, so they can focus on their applications logic. In this way, it is possible to achieve the abstraction of server management (provisioning, configuration, scaling, etc.) so that users, in this case developers, can focus on the aforementioned logic.

2. Related jobs

The advantages and disadvantages of migrating HPC applications to the cloud have been mentioned in previous paragraphs. However, the time and effort required to configure the virtual resources may be greater than the actual time and effort spent doing the calculations. On the other hand, if the serverless paradigm is used it will be possible to have more granular control over the service provided, leaving infrastructure management in the hands of the cloud provider.

In [9], a systematic mapping of 89 use cases has been carried out where the serverless paradigm was applied to solve mostly HPC-related problems. But there is little information on a performance comparison between the applications running on the serverless paradigm versus the same applications running on a traditional cloud infrastructure, in which a behaviour analysis can be done to later decide which the best solution to run HPC applications is.

This work uses [10] as a starting point and deepens the research tasks based on [11], [12], [13] among others in recent years, in which it has been explored and evaluated the performance of the use of serverless in HPC applications. While these studies show that serverless is easy to use and inexpensive, its effectiveness over the conventional approach to cloud applications has not yet been quantified.

3. Work Scenarios

Three scenarios are proposed: two main scenarios and another secondary one; in which it will be evaluated: Contextualization time, Execution time and Price. The implementations and study were carried out in the Google cloud (Google Cloud Platform).

The first evaluation scenario is the "traditional" cloud. This scenario is made on a resource that functions as PaaS for which Datproc will be used, that allows to operate a cluster with Spark of 4 nodes, 3 workers and 1 master, with a maximum of 8 vCPUs in all. On the cluster the interfaces provided by Google are used to operate with the cluster, mainly the REST resource to send jobs via HTTP and the Jupyter notebook.

The second scenario runs on Cloud Functions, it consists of 4 nodes, 3 workers and 1 master to make an equivalence with a cluster of 3 workers and 1 master of the Dataproc configuration. The master node will be responsible for triggering the worker function 3 times to boot 3 instances "simultaneously" and then collect the results obtained by each of those instances, to figure out a unique result that will be the final result of the problem to solve. To solve each part of the problem, the worker function will use Pandas to work the data file as a dataframe.

The third scenario is a combination of the previous PaaS (Dataproc) scenarios and "on-demand" FaaS (Cloud Functions) cluster. Through an event, in this case an HTTP request, a function is executed which, in turn, generates a template for automatic cluster generation that is sent to the Dataproc API. After that cluster creation, you start executing processing and then shut down and remove the cluster.

4. Problem to solve and configuration behaviour

The problem to be solved is to determine, based on historical data obtained by sensors from 2009 to 2022, the time of day when the amount of CO (carbon monoxide) is lowest. For this purpose, a dataset from open data provided by the government of Buenos Aires on the site https://data.buenosaires.gob.ar/dataset/calidad-aire is used. The dataset has 110,000 records (one record for each time of day, from 2009 to 2022 in Centenario, Córdoba and La Boca cities). It was performed a record cleaning without data, after which it was obtained a dataset with 70,000 records.

To make a valid comparison, the resolution of the problem must be compared with two equivalent configurations. The minimum resource configuration for each node in Dataproc is 1 vCPU and 3.5GB of RAM; therefore, the evaluation will be made on customized machines with these resources on the first and third configuration, and on instances of 2GB of RAM and 1 vCPU on the second configuration since it is an even number of resources that is compared despite there being a difference in RAM.

4.1 Scenario 1: PaaS setup

The aspects evaluated in this configuration are considered once the cluster is in operation, the time from when the cluster is created until it shuts down will be taken into account in scenario 3 that corresponds to the PaaS Configuration with executing function.

Execution time. To obtain an accurate execution time, the average execution time of 10 jobs is considered (see Fig.1), where the resulting average time is 56 seconds.

Estado	Región	Tipo	Clúster	Hora de inicio	Tiempo transcurrido
Completado	us-central1	PySpark	cluster-3w	9 oct 2022 17:13:41	59 s
Completado	us-central1	PySpark	cluster-3w	9 oct 2022 17:11:33	1 min 2 s
Completado	us-central1	PySpark	cluster-3w	9 oct 2022 17:09:50	55 s
Completado	us-central1	PySpark	cluster-3w	9 oct 2022 17:07:48	57 s
Completado	us-central1	PySpark	cluster-3w	9 oct 2022 17:05:32	53 s
Completado	us-central1	PySpark	cluster-3w	9 oct 2022 17:03:51	54 s
Completado	us-central1	PySpark	cluster-3w	9 oct 2022 17:02:23	57 s
Completado	us-central1	PySpark	cluster-3w	9 oct 2022 17:00:50	55 s
Completado	us-central1	PySpark	cluster-3w	9 oct 2022 16:58:58	57 s
Completado	us-central1	PySpark	cluster-3w	9 oct 2022 16:55:50	55 s

Fig.1. Execution time of 10 jobs.

Price of the solution. Having the necessary resources provisioned only involves solving the problem and it is the price per second of having an n1-standard-1 instance running: 0.00001319444 U$D. If the cluster consists of 4 instances of this type and is running for 56 seconds, the total cost of the 4 instances in Cloud Engine is: 0.00001319444 U$D / s * 56 s * 4 = 0.003166664 U$D. If the cost of the Dataproc fee is added, the final cost of having the cluster running for 56 seconds is: 0.0006222222 U$D + 0.003166664 U$D = 0.00378888884 U$D.

4.2 Scenario 2: FaaS Configuration

What it has been observed in this configuration covers all the aspects, since they are measurable from it and it does not require another particular scenario for an accurate measurement.

Contextualization time. A new function instance is started in two cases: When the function is deployed or when a new function instance is automatically created to scale up (vertically) to the load, or occasionally, to replace an existing instance. Starting a new function instance involves loading the runtime environment and code. Requests that include the launch of function instances, called cold starts, may be slower than those routed to existing function instances. The occurrence of an error in the execution of the code implies a restart of the instance and therefore its corresponding cold start.

In Fig. 2, you can see the difference in execution times between a hot and a cold instance, you can see that the light blue instance is a hot instance because the INIT log does not appear, in addition to the fact that the execution takes only 462ms. Although in the three, the execution of the function took approximately 450 ms, it can be seen that in the last two ones, since the initialization time is also added, the final execution time is 3322 ms and 3378 ms respectively.

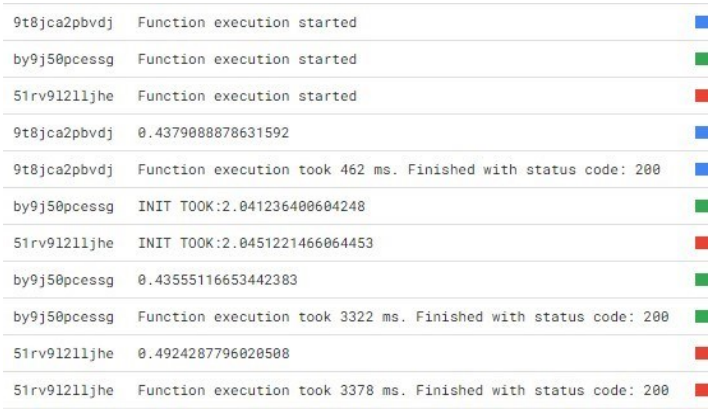


Fig. 2. Initialization time of a function instance.

The hot instance corresponds to the same instance that was initialized when updating the function, so the call of the first worker is routed to that instance, which of course also had its initialization time, but in a previous execution as shown in Fig.3.

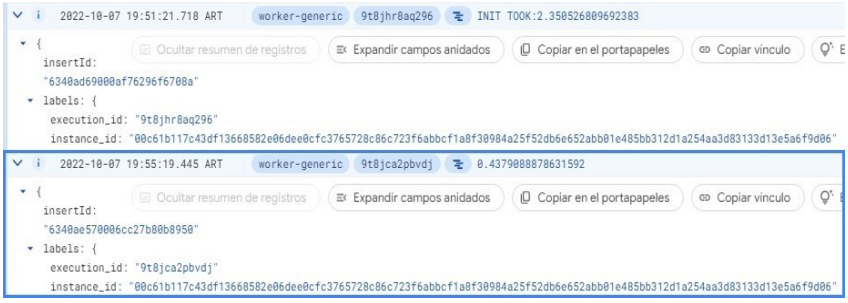


Fig. 3. Detailed logs showing that the execution of the light blue instance corresponds to a previous initialization of the instance.

The contextualization time of the functions, calculated as the average of 12 initializations, resulted in: 2.02986383438 seconds, for this case.

Execution time. Since contextualization time is different for both hot and cold instances, the execution time is calculated for both types of instances. To obtain an execution time, 10 executions of the master function are considered.

Hot run executions. Hot run executions are those that already have the execution environment and the code loaded, so there is no contextualization time in them. Next, it will be described what was observed when solving the problem with hot functions.

- Master instances: After carrying out 10 executions of the master function, the average execution time obtained is 295.3 ms.
- Worker instances: The execution time for the worker instances was calculated with the time resulting from each execution in the total number of worker instances executed by calling the master instance 10 times. The time resulting from the

calculation of the average in 30 instances is 177.4 ms.

Cold run executions. Cold executions require an extra contextualization time to the time necessary for the execution of the function.

- Master instances: it was obtained by averaging the times of 10 executions carried out, and the resulting time was 4423 ms.
- Worker instances: it was calculated from the 30 executions carried out by the master function. The time resulting from the calculation of the average in 30 instances is 2701 ms. This value considers the initialization time of the cold function, removing the initialization time from this equation, the average final execution time of the 30 executions is: 0.38844459056 s.

Price of the solution. The price calculated for the FaaS solution takes into account the runtime of the cold instances. In Cloud Functions, the price of having each worker instance running for 2700 milliseconds is charged to the nearest 100 milliseconds: 27 * 0.000005800 USD = 0.0001566 USD. The solution to the problem requires 3 worker instances and 1 master instance, so the final price is: 0.0001566 U$D * 3 +

0.000012501 U$D = 0.000482301 U$D

4.3 Scenario 3: PaaS configuration with executor function

This configuration has as its main objective to measure the times involved in the contextualization and shutdown of the cluster, it is a configuration that complements what cannot be observed in scenario 1.

Contextualization time. An important aspect to take into account is that the cluster does not start working at the time the job starts executing, but rather it has a provisioning and contextualization time at the beginning, another one when it is turned off, during which it cannot be used, and also to be taken into account at the time it is charged.

As the objective of this configuration is to know the time dedicated to provisioning, contextualization of the cluster and next shutdown after processing the jobs, 10 executions of this configuration are carried out to obtain, on average, what is the time dedicated to these processes that go unnoticed when running a cluster.

- Cluster startup and resource provisioning time: 228 s = 3m48s
- Cluster removal time: 61 s = 1m01s

Making a sum of these times and the average execution time of a job (56 s) obtained in the first configuration results in a total time of: 4m49s + 56s = 5m45s

Execution time. To know the final execution time the cluster takes to solve the problem considering provisioning, contextualization, processing and next deletion, the average of 10 executions was calculated. The total average duration of the execution of the cluster obtained was: 340.5 s = 5m40s

In other words, considering the average execution of a job in the PaaS Configuration of scenario 1 and the contextualization and cluster removal times of the current Configuration, it coincides with a 5-second error since it is an average of the total

execution time of the cluster with the sum of the provisioning, running, and removed averages.

Another aspect to take into account is that this time is the time required one to create, provision, process, stop and delete. Once the cluster is provisioned, since creation is not required, the startup time decreases a significant portion of the initial time required.

Price of the solution. The price per second of having an instance of type n1-standard1 running is $0.00001319444. If the cluster is on for 5m 40s on average and consists of 4 instances, then the final price of the four instances on Compute Engine is 0.00001319444 U$D / s * 340 s * 4 = 0.01792333318 U$D. To this situation the Dataproc fee must be added, the final price of running a cluster for 340 seconds is 0.00377777777 U$D + 0.01792333318 U$D = 0.02170111095 U$D

4.4 Comparison of what was observed

When solving the problem with the different configurations, important differences were observed in the aspects studied.

Contextualization time. When comparing contextualization times of both configurations (see Fig.4), it is observed that the PaaS Configuration takes almost 5 minutes, adding the cluster provisioning and shutdown time after processing the job. On the other hand, the FaaS Configuration, when considering cold runs, have a contextualization time of 2 seconds.

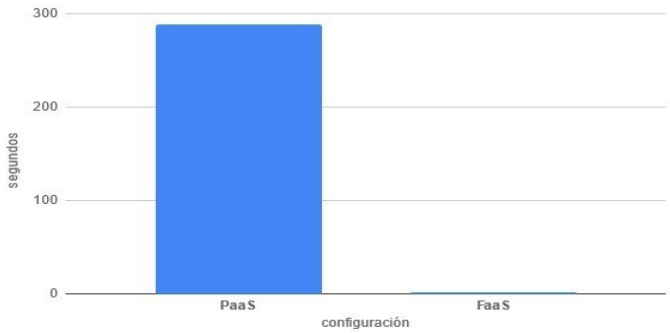


Fig.4. Contextualization time of each configuration.

Execution time. The PaaS Configuration requires 56 seconds on average, while the FaaS Configuration requires 366 milliseconds, being this less than 1% of the time that it takes the first configuration (see Fig.5). Also, the time required by FaaS is mainly dedicated to reading the file from storage.

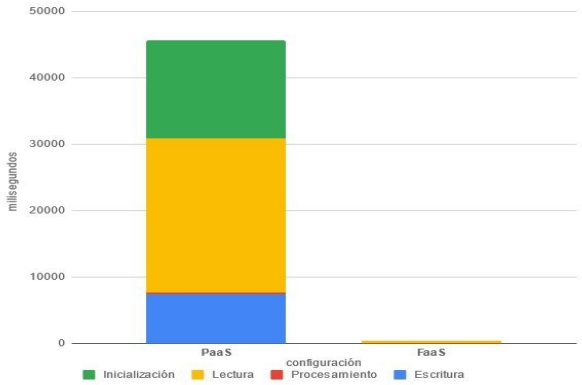


Fig.5. Runtime with stacked times.

Price. Finally, price is the aspect where there is a notable difference in using each of the configurations. As above, the price of PaaS Configuration exceeds the price of FaaS Configuration.

4.5 FaaS in detail

Next, some aspects and limitations of FaaS will be described, in addition to making some comparisons with some aspects of the scenario traditionally used in order to conclude if it is possible to build HPC environments based on FaaS and to what extent it would be feasible to do so.

Function execution time. Execution time is a key aspect for the resolution of a problem and this aspect in FaaS is limited; the maximum time that a function can be running is 540 seconds in GCP and varies depending on the provider.

Memory limit. Each instance of a function has a memory limit that is set when configuring it, function memory capacity can go from 128MB to 8GB.

To evaluate the behaviour of the functions in an HPC environment, it is necessary to take this aspect into account. The evaluation started by processing 2GB (56,999,943 rows) with each instance within the 1 master and 3 worker configurations, that is, each of the 3 worker instances read a 2GB file and processed it. From this first evaluation, it turned out that from 11 processes that were carried out (66 GB of data) only 3 of them responded correctly, that is, 27% of the total number of executions carried out (see Fig.6).

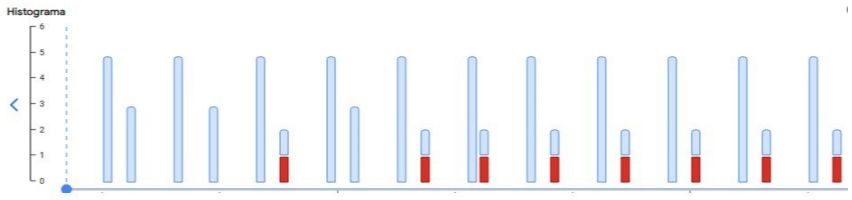


Fig.6. Percentage of correctness of the configuration.

Later on, the capacity of the worker separated from the complete configuration is evaluated to know the capacity of the running instance separated from the configuration. To do this, the percentage of executions that responded correctly with different loads was evaluated for 10 minutes, to know the point at which the percentage of correctness is 100%. The loads used, from highest to lowest, were 2 GB (56,999,943 rows), 1.3 GB (37,205,301 rows), 1 GB (29,999,971 rows), 500 MB (14,999,986 rows).

That is, for this particular problem with the processing that is done and the libraries that you use, for a function of 8 GB and 2 vCPU per instance, the optimal file size is 500 MB.

Higher scales and elasticity. Considering that the optimal file size for an 8 GB function is 500 MB, for each configuration of 1 master and 3 workers 1.5GB is processed.

As FaaS are self-scalable in terms of the number of received requests, it is possible to execute up to 10,000 instances simultaneously. For this case, 60 instances of the master function were executed simultaneously and, as it can be seen in Fig.7, the number of running instances can go from 0 instances to 60 almost instantly; so if each configuration of 1 master and 3 workers processes 1.5 GB, with 60 runs 90 GB are processed.



Fig.7. Number of instances of the master function running concurrently.

Large amount of data processed in a limited time. As seen above, 60 instances of the master function were executed, which involved processing 90 GB in all. The important thing about this is that, since FaaS runs in a different environment than the rest of the other executions of the same function, executions can overlap and manage to process large amounts of data in a short period of time. In Fig 8., it is shown that the average time taken to execute each processing in each execution is 29 seconds.

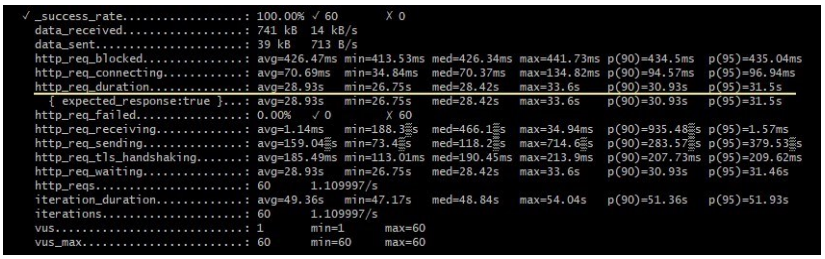


Fig.8. Duration of http request to master function.

That is, if the average execution took 30 seconds, and 60 simultaneous executions were launched, as each execution processes 1.5 GB, 90 GB of data was processed in 30 seconds.

Costs of the processing performed. The price per 100 ms of running an instance of an 8 GB function is 0.000006800 U$D. If each instance ran on average 30 seconds, the price per instance is 0.000006800 U$D * 300 = 0.00204 U$D. If 60 instances of the master function were executed and each master function executes 3 instances of the worker function, then the total cost for processing 90 GB in 30 seconds is 0.00204 U$D * 60 * 3 = 0.3672 U$D. Adding to this, the cost of running the 60 master instances that are 128MB the total price for running the environment for 30 seconds is 0.3672 U$D + 0.004158 = 0.371358 U$D

5. Conclusions

It is possible to build an HPC environment in Functions as a Service, but depending on the problem to be solved, it may or may not be a viable solution.

Positive points for which it could be viable to build an environment based on Functions:

- Fast processing of medium loads.
- Elasticity, allowing large amounts of data to be processed with multiple instances running simultaneously.
- Saving on costs.
- Automation and integration with external services.

Negative points by which Functions as a Service are not a reliable alternative to build an HPC environment:

- While the user does not take over the infrastructure, there is no actual autoscaling, he must know approximately the number of instances to run.
- With little flexibility, making changes to the processing code could cause the environment to require an increase in resources or be destroyed if it reaches the maximum.

- Inability to make an analysis of the data; an external application is required to view the results.
- Technically it works only for weakly coupled problems and specific cases.
- There are limitations of time and memory imposed by technology.
- It takes extra effort to develop the environment to solve a problem with some degree of coupling.
- Limitations in the amount of data that can be processed (5 TB) considering the limit of 10,000 running instances.
- The possible cost reduction translates into the rigidity of the environment, its development, maintenance and its possible disablement in case of a change in the characteristics of the problem to be solved.
- Limited customization of instances.

For all these aspects, depending on the problem that is sought to be solved, it may be feasible to build the environment based on FaaS. The most important benefits are the speed and elasticity of the functions, which require a loosely coupled or zerocoupling problem of low-medium load (up to 1TB) to work optimally. The monetary benefit can be considered, but it is not the main thing that should be considered when building an HPC environment.

References

1. Farhan, L., Kharel, R., Kaiwartya, O., Quiroz-Castellanos, M., Alissa, A., & Abdulsalam, M. (2018, July). A concise review on Internet of Things (IoT)-problems, challenges and opportunities. In 2018 11Th International Symposium On Communication Systems, Networks & Digital Signal Processing (CSNDSP) (pp. 1-6). IEEE.
2. Medel, D., Murazzo, M. A., Molina, A. L., Sánchez, F., Cornejo, M., Rodríguez, N. R., ... & Piccoli, M. F. (2019). La Computación de Alta Performance como soporte a los sistemas altamente distribuidos. In XXI Workshop de Investigadores en Ciencias de la Computación (WICC 2019, Universidad Nacional de San Juan).
3. Barrionuevo, M., Escalante, J., Lopresti, M., Lucero, M., Miranda, N. C., Murazzo, M. A., & Piccoli, M. F. (2020). Solución de grandes problemas aplicando HPC multitecnología. In XXII Workshop de Investigadores en Ciencias de la Computación (WICC 2020, El Calafate, Santa Cruz).
4. de Souza Cimino, L., de Resende, J. E. E., Silva, L. H. M., Rocha, S. Q. S., de Oliveira Correia, M., Monteiro, G. S., ... & de Castro Lima, J. (2017, November). IoT and HPC integration: revision and perspectives. In 2017 VII Brazilian Symposium on Computing Systems Engineering (SBESC) (pp. 132-139). IEEE.
5. Biswas, A. R., & Giaffreda, R. (2014, March). IoT and cloud convergence: Opportunities and challenges. In 2014 IEEE World Forum on Internet of Things (WF-IoT) (pp. 375-376). IEEE.
6. Añel, J. A., Añel, J. A., Montes, D. P., Iglesias, J. R., & Romano. (2020). Cloud and Serverless Computing for Scientists. Springer International Publishing.
7. Malla, S., & Christensen, K. (2020). HPC in the cloud: Performance comparison of function as a service (FaaS) vs infrastructure as a service (IaaS). Internet Technology Letters, 3(1), e137.

8. Baldini, I., Castro, P., Chang, K., Cheng, P., Fink, S., Ishakian, V., ... & Suter, P. (2017). Serverless computing: Current trends and open problems. In Research advances in cloud computing (pp. 1-20). Springer, Singapore.
9. Eismann, S., Scheuner, J., Van Eyk, E., Schwinger, M., Grohmann, J., Herbst, N., ... & Iosup, A. (2020). A review of serverless use cases and their characteristics. arXiv preprint arXiv:2008.11110.
10. Rodríguez, N. R., Murazzo, M. A., Medel, D., Parra, L., Molina, A. L., Sánchez, F., ... & Vargas, L. (2021). Procesamiento paralelo sobre arquitecturas serverless para tratamiento de datos provenientes del IoT. In XXIII Workshop de Investigadores en Ciencias de la Computación (WICC 2021, Chilecito, La Rioja).
11. Niu, X., Kumanov, D., Hung, L. H., Lloyd, W., & Yeung, K. Y. (2019, September). Leveraging serverless computing to improve performance for sequence comparison. In Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics (pp. 683-687).
12. Spillner, J., Mateos, C., & Monge, D. A. (2017, September). Faaster, better, cheaper: The prospect of serverless scientific computing and hpc. In Latin American High Performance Computing Conference (pp. 154-168). Springer, Cham.
13. Chard, R., Skluzacek, T. J., Li, Z., Babuji, Y., Woodard, A., Blaiszik, B., ... & Chard, K. (2019). Serverless supercomputing: High performance function as a service for science. arXiv preprint arXiv:1908.04907.

Coordinadores:
- Horacio Villagarcia Wanza (UNLP)
- Emanuel Frati (UNdeC)
- Jorge Ierache (UM)

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

XIV Workshop de Procesamiento de Señales y Sistemas de Tiempo Real

> Full Papers

WPSSTR

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

ADS-B Collaboration with SSR

Oscar Bria[0000-0002-0001-4248] and Javier Giacomantone[0000-0001-9362-9323]

Research Institute in Computer Science (III-LIDI) - School of Computer Science
National University of La Plata - Argentina
onb@info.unlp.edu.ar

Abstract. In an air traffic surveillance station with SSR and ADS-B, the information generated by each sensor can be used to improve the performance of the other. In addition to improving the predictive performance required by Roll-Call interrogations, ADS-B can help to improve SSR performance in the vicinity of the cone of silence, allow passive acquisition and help to identify false SSR targets generated by reflection.

Keywords: ADS-B, SSR, Modo S, Collaboration among Sensors.

1. Introduction

Secondary Surveillance Radar (SSR) and Automatic Dependent Surveillance System by Broadcast (ADS-B) are two cooperative technologies for air traffic surveillance.

The SSR [1] independently determines the position of the aircraft which, in turn, is interrogated by the radar to gather complementary information. The SSR Mode S (Selective) [2] is the second generation of the SSR; allows selective interrogation, Roll-Call interrogations, using an exclusive 24-bit identification code for each aircraft [3,4]. SSRs typically have a position update rate between 4 and 10 seconds depending on the rotation speed of the antenna.

ADS-B [5] is dependent on Global Navigation Satellite Systems (e.g., GPS) for position determination, which is emitted along with other information spontaneously by the aircraft and received on the ground by a system simpler than a radar. ADS-Bs typically have a position update rate of around 1 second [6,7].

Nowadays it is common to find an SSR with Mode S and an ADS-B installed at the same site, therefore, with mostly overlapping coverage. In [8] a method of using the ADS-B tracker was presented to improve the prediction needed by SSR Mode S, when both sensors are located in the same place. One of the motivations for proposing this collaboration is that, as already shown, the ADS-B position update rate is significantly lower than that of SSR.

In the process of developing an integrated collaboration system between an ADS-B and a SSR, it is possible to include other improvements besides the correlation of the position of the aircraft reported by both sensors. The limitations imposed by the so-called cone of silence of the SSR coverage can be mitigated. The performance of the reflection detection and elimination system of the SSRs can also be improved.

2. Collaborative Trackers

In [8] a simple scheme of collaborative trackers [9,10] (see Figure 1) was presented for the prediction of the aircraft position, necessary for the timing of the SSR Roll-Call interrogation for the next antenna visit.

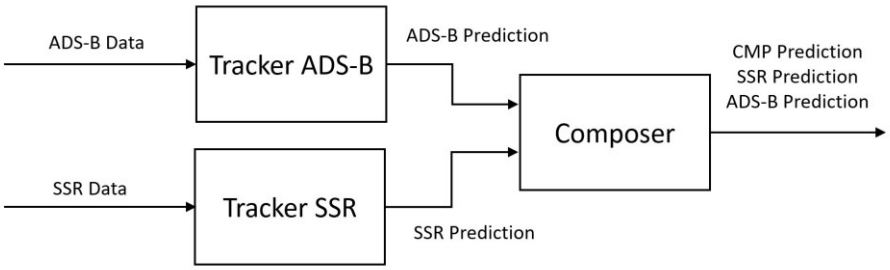


Fig.1: Block Diagram of the Collaboration between Trackers.

The block named Composer in Figure 1 calculates the average prediction weighted by the variances according to the following calculation [11]:

$$P = \left[P_1^{-1} + P_2^{-1}\right]^{-1} \quad (1)$$

$$\hat{x} = P\left[P_1^{-1}\hat{x}_1 + P_2^{-1}\hat{x}_2\right] \quad (2)$$

Where P_i are the individual covariances of each sensor prediction and $\hat{x}_i$ are their predictions. While P and $\hat{x}$ are the respective values of the composition.

The compose (CMP) prediction, given by the pair $[\hat{x},P]$, is statistically better[1], than the SSR prediction [12,13]. That contributes to better estimation, more efficient use of polling scheduling resources, and better performance of mode S in various respects [8].

3. Cone of Silence and Passive Acquisition

Radar coverage is the volume of airspace scanned by a radar or radar network [14]. The coverage volume of a surveillance radar is the volume generated by the revolution, around the antenna rotation axis, of the surface delimited by the following bounds (typical values are shown in brackets):

1. The minimum elevation angle from the horizon [0°].
2. The maximum elevation angle from the horizon [45°].
3. The minimum slant distance from the center of the antenna [0.5 NM].
4. The maximum slant distance from the center of the antenna [150 NM].
5. The maximum altitude with respect to the ground [FL 1000].

[1] The magnitude of the composition variance, P, is always less than the magnitude of any of the individual variances, P_i.

The solid generated by revolution includes a blind volume in the shape of an inverted cone contained between the maximum elevation angle and the axis of rotation of the antenna called the cone of silence (CoS)[2]. The CoS is produced by the effect of the shape of the irradiation pattern of the radar antenna, which in the case of SSRs has a square cosecant pattern [15]. Figure 2 shows the CoS in schematic form. For example, an aircraft flying at a cruising altitude of FL 300 (approx. 10 Km) will enter the CoS at 5.4 NM from the radar (approx. 10 Km) if the coverage reaches 45° in elevation. The SSR does not detect the aircraft while inside the cone, the track will be lost and it must be acquired again when the aircraft leaves the CoS, if at all. The re-acquisition process involves one or more All-Call queries.

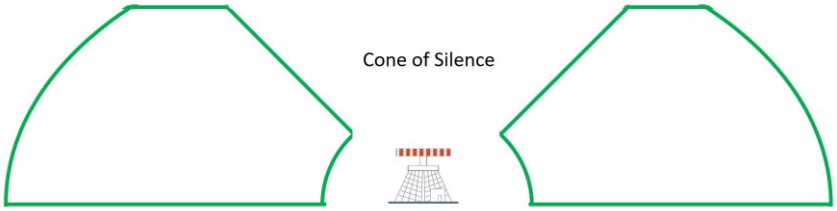


Fig.2: Vertical-Central Section of the Coverage Volume (not to scale).

The ADS-B collaboration function with the SSR improves the acquisition process of aircraft with squitter and mode S transponder at the exit of the cone of silence, avoiding the delay imposed by the exchange of interrogations and All-Call responses as usual in the active acquisition of the SSR.

With the help of the data supplied by ADS-B, the output of the SSR cone of silence is anticipated in such a way that a passive acquisition is carried out that allows the first interrogation of the SSR outside the cone of silence is directly a Roll-Call interrogation.

This function must include an active tracker based on ADS-B information for every possible aircraft that may exit the SSR's CoS and also must anticipate departure, early enough for the SSR to schedule a Roll-Call interrogation at the appropriate time.

Figure 3 is a diagram with details of the working environment of the collaboration within the CoS according to the following detail:

1. The dotted line above represents a trace of the responses or possible SSR responses given by an aircraft moving away from the radar heading East. The black dot corresponds to the situation where the radio range is not enough to process responses at that distance (which on the other hand may not be produced). The two blue dots correspond to responses not processed by the SSR because they are beyond its Surveillance Coverage (vertical green line farther to the right). The next two green dots correspond to responses processed by the SSR but not reported to the console, which correspond to All-Calls responses that are used to initialize the predictor for the next round. The two red dots correspond to responses processed and reported to the console, which are in correspondence with the responses to

[2] Another blind zone can also be generated if the angle of minimum elevation is above the horizon. This can occur in the azimuthal sectors including obstacles.

Roll-Call selective interrogations.

2. The central line represents the squitter emissions of the same aircraft moves away from the SSR moving from West to East, and which may or may not be processed. The key to colors is similar to the previous one.
3. The line below corresponds to the responses or possible responses of the aircraft to the SSR, when it works in a composite way with ADS-B. The color key is the same as before.

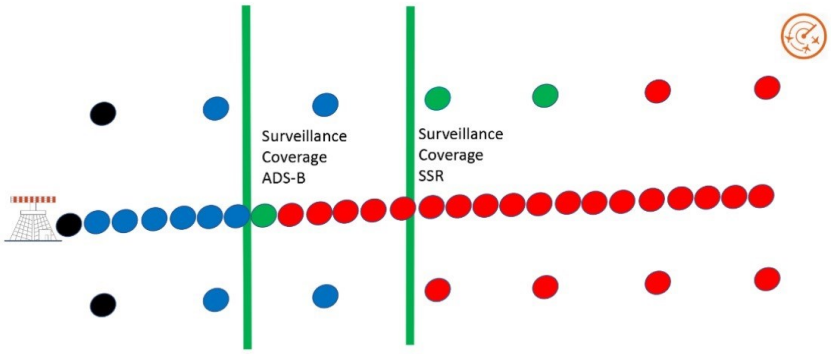


Fig.3: Detail of the Collaboration in the Cone of Silence.

For the third trace, it is observed that the two points immediately to the right of the second vertical green line are red. This is so because the SSR received information from ADS-B so that these points correspond to Roll-Call responses. This is possible if the ADS-B information arrives in a timely manner.

The details of how forecast anticipation is carried out will depend on how accurately the limits of the cone of silence overlapping with the minimum range are known or defined together with the presence or absence of a surveillance coverage map, and of the forward forecast accuracy that can be achieved with ADS-B data (see fixed point forecast at [11]). Due to the indeterminacy in the knowledge of the limits, superfluous or, on the contrary, few Roll-Call interrogations could be programmed.

The improvement of the order of two turns depends, in terms of distance, on the speed of the aircraft and the rate of rotation of the antenna. The improvement in the distance is of the order of 2 NM considering typical values of both speeds for an en-route traffic radar, as observed in Equation (3). The improvement does not depend on the altitude at which the aircraft leaves the CoS, but is still significant for the upper limit of typical coverage.

$$2\text{turns}\ \frac{450\text{Knots}}{1\text{turn}/8\text{seconds}} = 2\text{NM} \qquad (3)$$

In practice, it is necessary to distinguish between the radio frequency range (distance up to which radio frequency signals can be processed) and the surveillance coverage which is set operationally for each sensor. Figure 4 shows the radio range and surveillance coverage for the SSR, further showing the flight situations that should be taken into account.

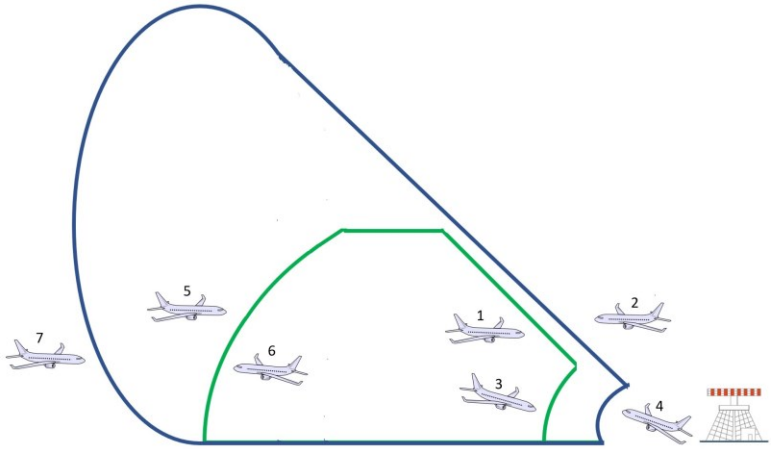


Fig.4: Radio Range and Surveillance Coverage.

1. Entering Cone of Silence.
2. Leaving Cone of Silence.
3. Landing in Blind Zone.
4. Taking off from Blind Zone.
5. Entering Surveillance Coverage.
6. Leaving Surveillance Coverage.
7. Entering Radio Range.

Case number 5 (Entering Surveillance Coverage) in Figure 4 can be treated as case number 2. (Leaving Cone of Silence) if there is ADS-B coverage beyond Surveillance Coverage. Also, in this case it is possible to acquire the trace through a passive acquisition using a technique similar to the one shown.

4. Reflections

Although SSR and ADS-B are both cooperative surveillance technologies, a very important distinction between them is that the former estimates the position (minus altitude) independently and the latter dependently [16]. Therefore, what in principle is a disadvantage for the latter can be helpful or essential to determine if a response is real or comes from a reflection, as shown below.

False detections occur due to reflection of electromagnetic waves by obstacles, both for interrogations and for responses from an SSR. Figure 5 shows the geometry of the reflections.

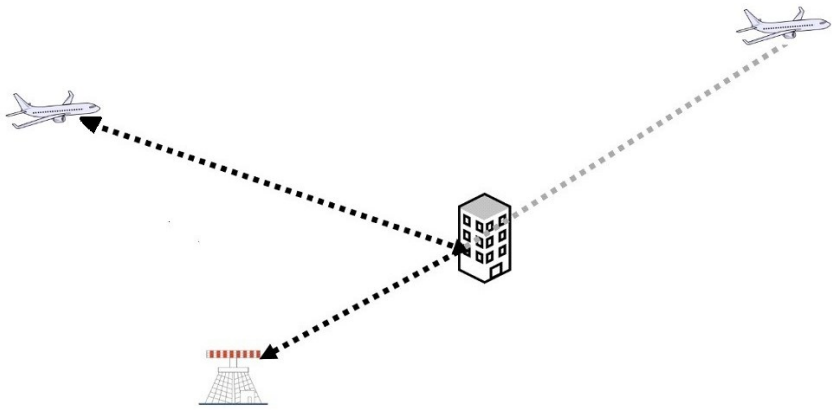


Fig.5: Geometry of Reflections

In Figure 5, the actual location of the aircraft is on the left, but when the radar has interrogated in the direction of the building it has generated a reflected interrogation which has also generated a reflected response which is received by radar. The SSR computes the position of the aircraft and places it in the position to the right.

Below are described several particular reflection situations in which ADS-B can contribute in varying depth.

1. When the interrogation mode is A/C the reflections can be identified through a well known procedure. If, for example, two aircraft with the same A code are detected in one turn of the antenna, it can be assumed that the aircraft that is at a greater distance from the radar is actually a reflection. In addition, the position of the reflector can be calculated, which is supposed to produce reflections for any other similar situation. Thus, it is convenient to survey a map of the reflectors present in the vicinity of the radar.

   Assuming v is the distance at which the real aircraft is above azimuth α and r is the distance at which its reflection is seen above azimuth β [3]. Then, the distance to the reflector R and the orientation of the normal vector to the reflecting plane $\perp$ can then be calculated. See Figure 6 in which the guidelines of geometric optics are followed and the small black segment represents the reflecting plane.

[3] α and β are measured with respect to the North turning to East.

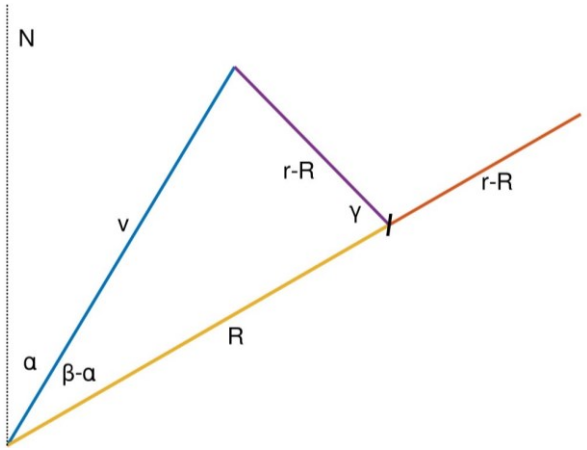


Fig.6: Geometry for calculating the Location of the Reflector.

Applying the laws of cosines and sines in the triangle of Figure 6,

$$R = \frac{1}{2}\frac{r^2 - v^2}{r - v\cos(\beta - \alpha)} \tag{4}$$

$$\perp = \frac{1}{2}\arcsin\frac{v\sin(\beta - \alpha)}{r - R} + \beta + \pi \tag{5}$$

The survey of the map of permanent reflectors for the airways of a radar site is part of the site study that must be carried out for each SSR installation [17]. In addition to the permanent reflectors, mobile reflectors that eventually appear should be temporarily included on the map. An obstacle can operate as a single or several reflectors in different planes. Relatively large obstacles, in addition to producing reflections, can produce blind zones at low altitudes as mentioned previously.

2. The S Roll-Call interrogation mode is, in principle, immune to reflections because interrogations are directed to a single aircraft known to be in the azimuthal direction towards the main beam of the radar antenna is pointing. What could happen is that through a reflector, during the All-Call interrogation period, the identifier S of an aircraft is acquired and, thus, it goes on to be interrogated Roll-Call through the reflector in the next few turns. Indeed, this condition could be detected, as for mode A/C, by the repetition of the identifier S and also the furthest detection is the false one[4].

[4] In this case, the reflection identification is more secure because it is known that the identifier S cannot be repeated.

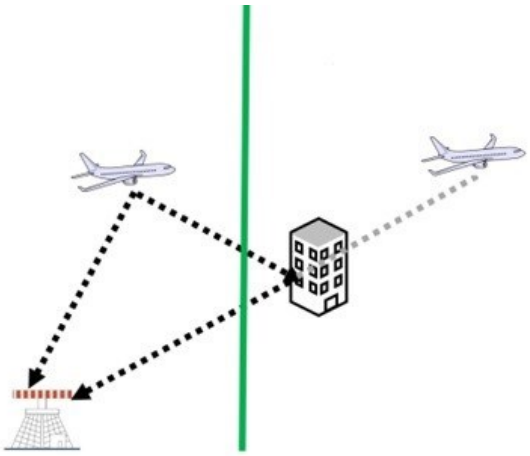


Fig.7: Singular Case of Geometry of Reflections.

3. Figure 7 shows a singular case of a mirroring situation. Here, the aircraft is within the CoS of the SSR and it is interrogated through a reflector that is outside the CoS. The aircraft responds for both a mode A/C interrogation and a mode S interrogation. In both situations there are no references to compare and determine that they are reflections. This is so because the same aircraft does not appear in any other azimuth of the turn. In both cases, the mistake is made by accepting the reflected aircraft on the right as existing. In other words, the traditional algorithm, only based on the SSR, does not distinguish the reflection as such for this geometry. The ADSB, which distinguishes the aircraft within the CoS, can contribute indicating that there is an aircraft that is responsible for a reflection and can contribute to the location of the reflector with a similar method to the one already explained.
4. ADS-B spontaneous emissions can also produce reflections similar to those of SSR radar, but are indistinguishable from direct emissions, except for arrival time, because ADS-B relies on the GNSS system for its location, and not on a measurement. Those refection situations must be solved by the ADS-B receptor itself or by post processing performed from the ground equipment.

5. Conclusions and Future Work

Additional features of ADS-B collaboration with SSR have been introduced when both sensors are at the same site. The contribution is significant for: ✓ Anticipate and simplify the acquisition of mode S tracks at the limits of the coverage.

✓ Allow and anticipate the acquisition of SSR tracks at the output of CoS.

✓ Reinforce and allow the identification of reflections that produce false targets in the SSR.

In those cases and in the case of the tracker, the integration of both sensors can be done by collecting data directly from the Asterix Cat.21 output of the ADS-B to help the Asterix Cat.48 generation processing of the SSR (see Figure 8).

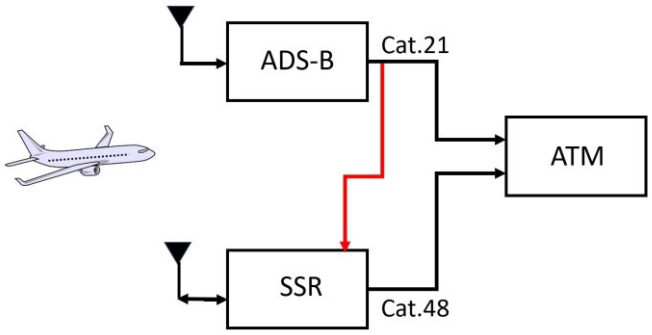


Fig.8: Asterix Collaboration Scheme.

The preliminary design must be finalized and its performance evaluated by simulating plausible scenarios, including simulated flights in holding patterns such as those existing in the vicinity of airports [18] and simulated flights within the SSR CoS [19].

In a detailed design, it is necessary to reconcile aspects of the protocol with the physical aspects, particularly in the areas close to the radar [20] and include functional aspects of the squitter as delayed transmission [21][5].

A posteriori, to the extent possible, tests should be implemented on the SSR and ADS-B installations of a real site [22], to collect statistics that can quantitatively verify the effective improvements provided by the functionalities

References

1. ICAO (International Civil Aviation Organization). Manual on the Secondary Surveillance Radar (SSR) Sytems. In: Doc 9684 AN/951 (2004)
2. Orlando, V., Drouilhet, P.: Functional Description of Mode S Beacon System. In: Project Report ATC-42 Revision B, Lincoln Laboratory, MIT (1982)
3. Stevens, M.: *Secondary Surveillance Radar.* Artech House (1988)
4. ICAO (International Civil Aviation Organization). *Annex 10, Third Edition of Volume IV.* (2014)
5. ICAO (International Civil Aviation Organization). Technical Provisions for Mode S Services and Extended Squitter. (2008)
6. RTCA: Minimum Operational Performance Standards for 1090 MHz Extended Squitter Automatic Dependent Surveillane - Broadcast (ADS-B) and Traffic Information Services (TIS-B). In: RTCA DO-260B (2011)

[5] As a consequence of delayed transmission, Figure 3 should be modified accordingly.

7. EUROCAE (The Europena Organisation for Civil Aviation Equipment): *Technical Specification for a 1090 MHz Extended Squitter ADS-B Ground System.* In: EUROCAE ED-129B. (2016)
8. Bria, O., Giacomantone, J.: *Colaboración ADS-B en la predicción SSR.* In: Libro de actas del XXVIII CACIC - Congreso Argentino de Ciencias de la Computación. (2022)
9. Sun, J.: The 1090 Megahertz Riddle: A Guide to Decoding Mode S and ADS-B Signals. 2nd Edition. TU Delft OPEN Publishing (2021)
10. McDewitt, A.J.: A Tracker for Monopulse SSR. In: IEE Colloquium on State Estimation in Aerospace And Tracking Applications (1989)
11. Bar-Shalom, Y., Rong Li, X., Kirubajaran, T.: *Estimation with Applications to Tracking and Navigation*, John Wiley & Sons, Inc. (2001)
12. Welch, G., Bishop, G.: An Introduction to the Kalman Filter. In: TR 95-041, Department of Computer Science, University of North Carolina at Chapel Hill (2006)
13. Broookner, E.: *Tracking and Kalman Filtering Made Easy*, Wiley-Interscience. (1998)
14. Radartutorial.eu: *Fundamentos Radar. Cobertura Radar*. In: https://www.radartutorial.eu/01.basics/rb21.es.html (2023)
15. Balanis, C.: *Antenna Theory: Analysis and Design*, John Wiley & Sons (2015)
16. Skybrary.aero: *Surveillance. definitions*. In: https://skybrary.aero/articles/surveillance (2023)
17. EUROCONTROL.: European Mode S Station Functional Specification (EMS), edition 4.0, EUROCONTROL-SPEC-189 (2021)
18. FAA, Aeronautical Information Services: Aeronautical Chart Users Guide. In: aernav.faa.gov/userguide/20220714/cug-complete.pdf (2022)
19. Mariano, P., De Marco, P., Giacomini, C.: *Data Integrity Augmentation by ADS-B SSR Hybrid Techniques*. In: Integrated Communications Navigation and Surveillance Conference (2018)
20. U.S. Department of Transportation, Federal Aviation Adminitration Specification: Mode Select Beacon System (Mode S) Sensor. In: FAA-E-2716 (1983)
21. EUROCAE (The Europena Organisation for Civil Aviation Equipment): Minimum Operational Performance Specification for Secondary Surveillance Radar Mode S Trasnsponders. In: EUROCAE ED-73B. (2003)
22. Addullah, A., Ismail, A., Badron, K., Rashid, N.: Improving Radar Detection by Adaptation of Automatic Dependent Surveillance-Broadcast (ADS-B) Technology, In: Advance d Science Letters, Vol. 22 (2016)

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Determinación de la Orientación en Vehículos No Tripulados

Ignacio Zaradnik[1], Luciano Salvatore[1], Oreste Daniel Lupi[1], Diego Turconi[1], Juan Pablo Calabro[1]

[1] Departamento de Ingeniería Universidad Nacional de la Matanza Buenos Aires, Argentina
izaradnik@unlam.edu.ar, luciano.salvatore431@gmail.com

Resumen. El desarrollo de vehículos no tripulados presenta un gran número de áreas de aplicación. Una de estas áreas es el monitoreo de la calidad del agua de ecosistemas acuáticos, y dentro de ella la determinación del contenido de micro y nano plástico. Lo que se logra con la incorporación de un sistema de toma de muestras a un vehículo de superficie no tripulado. En estos dispositivos, la orientación es un factor importante, ya que en base a ella el sistema de control opera sobre los actuadores para lograr la trayectoria deseada. En el siguiente trabajo se presentan las pruebas realizadas con giróscopos y magnetómetros sobre un vehículo terrestre no tripulado, por encontrase el vehículo de superficie bajo construcción, para la determinación de la orientación. Los resultados obtenidos permiten determinar la orientación con un error aceptable, presentando el uso del magnetómetro la ventaja de la determinación estática de la orientación.

Palabras claves: Vehículos no tripulados, Ecosistemas acuáticos, Giróscopo, Magnetómetro.

1. Introducción

El contenido de micro/nano plásticos en el agua tienen un efecto sobre los seres vivos, ya sea por ingestión o por toxicidad. En el caso puntual de la concentración de estos en ambientes acuáticos, si bien es posible su detección, tal como se detalla en [1] o en [2-4], dichos métodos tienen la dificultad de no poder ser realizados en tiempo real y de ser muy costosas las campañas de muestreo y análisis. Un dispositivo adecuado para la toma de las muestras antes mencionado puede ser un pequeño vehículo de superficie no tripulado (USV por sus siglas en ingles), manejado a través de control remoto o parcialmente autónomo. En la actualidad existen diversos productos comerciales como los detallados en [5], así como diversos trabajos que presentan este tipo de solución [6-8], inclusive con la toma de muestras [9]. Los primeros poseen un costo elevado, mientras los segundos están pensados para aplicaciones que pueden no adecuarse a las necesidades propias y la documentación disponible no es adecuada para su reproducción.

Por lo tanto, como fruto de conversaciones con investigadores de la Fundación Argentina de Nanotecnología (FAN), quienes estaban interesados en la medición de estos parámetros, se planteó desarrollar un pequeño USV experimental autónomo y/o por control remoto para la medición de parámetros ambientales y la toma de muestras en ecosistemas acuáticos.

En el modo de funcionamiento autónomo, el control de la trayectoria depende de tres parámetros: la posición, el curso y la orientación. Es importante aclarar que el curso

y la orientación no son lo mismo, mientras que el curso nos indica en qué dirección se mueve el vehículo, la orientación nos indica en qué dirección esta apuntando el mismo, una simple explicación se puede encontrar en [10]. En función de la orientación, el sistema de control del vehículo opera sobre los actuadores a fin de mantener el curso que posee o cambiarlo, y de esta forma llegar de la posición actual a la deseada.

La posición y el curso del dispositivo puede ser obtenidas por medio de la incorporación de un módulo GNSS (siglas en ingles de Sistema global de navegación por satélite) a través de la lectura de la trama RMC (Datos de tránsito/GPS específicos mínimos recomendados por sus siglas en inglés) del protocolo NMEA [11]. Mientras que para la orientación es necesario el uso de sensores inerciales: acelerómetro, magnetómetro y giróscopo. A través del magnetómetro se obtiene la orientación por medio de la medición del campo magnético terrestre [12], mientras que el acelerómetro y el giróscopo pueden ser utilizados para compensar el movimiento del sistema y mejorar el cálculo de la trayectoria como se describe en [13].

2. Desarrollo

2.1 Arquitectura del Sistema

En la figura N°1 se puede ver la arquitectura planteada para el USV, en donde se tiene un sistema central y subsistemas para las distintas tareas a desarrollar. El subsistema de alimentación brindará la energía necesaria a cada uno de los elementos del dispositivo. El subsistema de radiocontrol recibirá las órdenes de movimiento para el posicionamiento inicial y rescate del dispositivo. Con la posición destino, determinada previamente o informada por el subsistema de telemetría, el sistema central de procesamiento determinará la trayectoria a seguir en función de su posición actual, obtenida por un módulo GNNS incluido en el subsistema de posición. El cual, además, cuenta con sensores inerciales que brindarán información para los ajustes de la trayectoria. Con la trayectoria determinada, se enviarán comandos al subsistema de propulsión para el manejo de los motores. Una vez que se llega a destino, se activará el subsistema de toma de muestra.

2.2 Hardware y Software empleado

A continuación, no se detallará todo el hardware asociado al dispositivo, sino sólo el que esté involucrado con las rutinas de posicionamiento y orientación.

Como sistema central de procesamiento se empleó la placa Orange Pi PC2 es una SBC (Single-Board Computer), la cual integra un procesador Quad-core de 64 bits

ARM-Cortex A53, 1GB de memoria RAM (DDR3), 2GB de memoria flash (NOR), interfaz para SD y múltiples interfaces de comunicación [14]. En lo que respecta a los sensores inerciales se empleó el dispositivo MPU 9250, el cual integra un giróscopo, un magnetómetro y un acelerómetro [15].

El sistema operativo embebido en la plataforma seleccionada es un Armbian [16], que no es otra cosa que un Debian (una de las distribuciones de Linux más populares) compilado con los requerimientos necesarios para ejecutarse sobre procesadores con

arquitectura ARM. Las rutinas fueron implementadas en C sobre un entorno Eclipse en PC, y luego se realizó una compilación cruzada (Cross-compilation) con ayuda de Linaro Toolchain ARMv7 [17] para embeberlas en la placa.

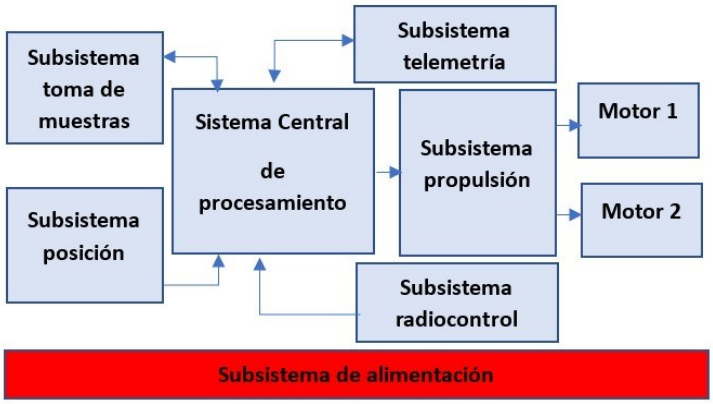


Fig. 1. Aquitectura de embarcación autonóma.

2.3 Pruebas iniciales

Partiendo de los ensayos previamente realizados sobre la placa Orange Pi, donde se logró obtener las lecturas de un sensor inercial conectado a la interfaz I2C (no detallados por cuestiones de extensión), la siguiente tarea consistió en lograr el correcto procesamiento de los datos para conseguir información útil de la rotación del dispositivo.

Para la determinación de la orientación en base al magnetómetro se consideraron las mediciones en los ejes “x” e “y” del mismo. Esto en base a que, si el eje “x” esta alineado con el campo magnético terrestre la medición en éste será máxima y cero en el eje “y”, mientras que si no lo está se medirá una intensidad en el eje “y”. El ángulo de desplazamiento se obtendrá como el arco tangente de la magnitud del eje “y” sobre el “x”.

Como consecuencia de que los problemas intrínsecos que poseen los magnetómetros, magnetización remanente a causa de su elevada sensibilidad y diferencia de sensibilidad entre ejes, fue necesario la implementación de una rutina de calibración similar a la empleada en teléfonos móviles. La misma consta en girar el sensor sobre su eje Z, mientras se toman muestras de los ejes X e Y, quienes se encuentran recorriendo todo su rango de valores posibles. Con cada nueva muestra se definen un nuevo valor máximo, mínimo y/o medio según corresponda al caso. Tras 5 segundos de rotación, con la información obtenida se procede a normalizar los valores de las futuras mediciones a realizar. Esto permite aumentar la precisión en la obtención del ángulo a coste de perder información sobre la densidad de flujo magnético, siendo esto último no influyente en el ensayo.

Las pruebas de medición orientación (ángulo respecto al norte magnético) fueron satisfactorias, la mayor precisión se encontró siempre que la brújula estaba en movimiento, llegando a adquirir una dirección final (tras acabar el movimiento) tal que

el error obtenido era menor a ± 1°. Al quedarse estático el sensor, se podían apreciar oscilaciones que alcanzaban los ± 7°, pero siempre alrededor de un valor medio previamente.

2.4 Ensayo

Para validar el uso de magnetómetro en la determinación de la orientación se planteó una prueba de odometría. Para este fin fue necesario contar con un vehículo autónomo o controlado por un usuario, sobre el cual estimar su posición y dirección. Si bien estaba previsto trabajar sobre un USV, las pruebas iniciales se desarrollaron sobre un vehículo terrestre no tripulado (UGV por sus siglas en inglés) constituido por un kit comercial de chasis circular con un juego de ruedas motorizadas de forma diferencial, sensores de proximidad laterales y distancia al frente, figura N°2. En donde, de toda la parte motriz y de detección se encarga de forma autónoma un microcontrolador, mientras que la Orange Pi se ocupa de registrar en intervalos regulares tanto la posición en coordenada "x" e "y", como la dirección en grados sexagesimales. El uso del kit comercial se debió a que el USV se encontraba en diseño y construcción.

En la figura N°3 se aprecia la rutina asociada al uso del magnetómetro para la determinación de la orientación. En ambas, la posición en "x" e "y" se determina como el producto de una distancia (obtenida por los encoders) y el coseno del ángulo determinado por el magnetómetro.

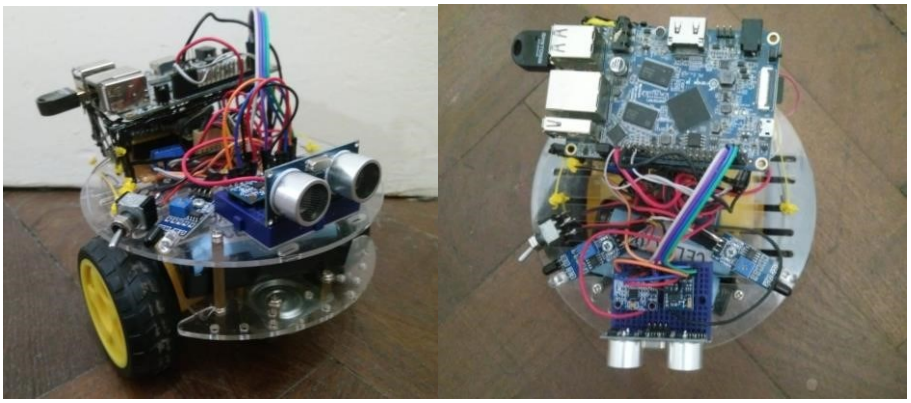


Fig. 2. Prototipo de vehiculo terrestre no tripulado implementado para prueba de rutinas.

Aprovechando la libertad de movimiento del autómata, se lo confinó a un cuadrilátero de aproximadamente 1 metro cuadrado para la realización de la prueba; el diámetro del robot que se ensambló para este fin es de 130mm con una distancia de pisada entre ruedas de 125mm. El procedimiento de ensayo fue el siguiente: tras encenderse el robot, comenzó a girar en su lugar durante 5 segundos para permitir que la rutina de calibración del magnetómetro ajuste el mismo. Transcurrido el tiempo, el robot se detuvo y rotó nuevamente hasta quedar en la dirección de 0 grados, con una tolerancia de ± 7°. Completado este proceso, se detuvo durante 3 segundos mientras las coordenadas eran transmitidas de forma inalámbrica para su comprobación.

Completada la transmisión, se avanzó en línea recta a unos 0,23 [m/s] hasta detectar una pared. Los sensores laterales corrigieron sólo lo necesario para avanzar paralelo a la pared, y el sensor frontal provocó un giro a la izquierda (si hay pared presente). De esta forma, recorrió toda la periferia del cuadrilátero y volvió a la posición original, trasmitiendo nuevamente las coordenadas.

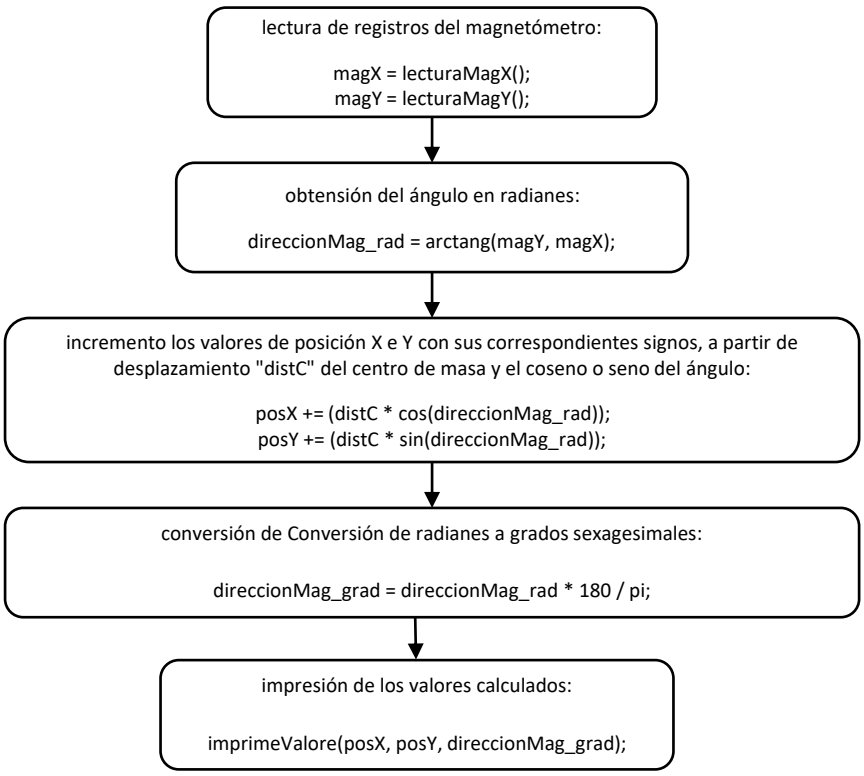


Fig. 3. Rutina del magnetómetro.

3. Resultados

El procedimiento se repitió varias veces. En cuanto a la estimación de coordenadas "x-y", se pudo identificar que al colisionar con una pared (falla del sensor o respuesta lenta de los motores) se producía el efecto de resbalamiento sobre las ruedas, lo que desplazaba mucho la coordenada de su posición real. En las oportunidades donde el robot era capaz de dar una vuelta completa sin incurrir en una colisión o sucesivas correcciones de rumbo, el error obtenido en "x" era de ± 1 punto, y el de "y" de ± 2 puntos, independientemente del sensor que se utilizaba, teniéndose en cuenta que un punto equivale a 130 mm (diámetro del robot).

4. Conclusiones

Se logró la implementación de rutinas para determinar la posición y la orientación en base a sensores inerciales. Si bien las pruebas realizadas no fueron sobre el dispositivo final (porque el mismo se encuentra en construcción), las mismas aportaron una valiosa experiencia al grupo de trabajo. Entre ellas se pueden nombrar: la familiarización con Linux embebido, los aspectos matemáticos asociados a la determinación de la posición y la orientación y las consideraciones de calibración asociadas al magnetómetro.

Como parte de los futuros trabajos se pretende implementar rutinas de filtrado Kalman para mejorar el error estático del magnetómetro y de compensación por la diferencia entre el norte magnético y el norte terrestre. Así mismo se pretende implementar estas rutinas sobre la embarcación propuesta y en conjunto con un módulo GNSS con el fin de poder desarrollar el control de trayectoria del vehículo de superficie no tripulado.

Referencias

1. Sergio Calvo Anglada.: Detección y Caracterización por Tamaños de Micro y Nanoplasticos en Muestras de Interes Ambiental. Universidad Zaragoza (2020)
2. M. C. Evans and C. S. Ruf.: Toward the Detection and Imaging of Ocean Microplastics With a Spaceborne Radar. En IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1-9, (2022)
3. D. G. Kotsifaki, C. Ripken and S. N. Chormaic.: Detection and analysis of microplastics in the subtropical ocean of Okinawa using micro-Raman Optical Tweezers. En 2021 International Workshop on Metrology for the Sea; Learning to Measure Sea Health Parameters (MetroSea), Reggio Calabria, Italy, pp. 158-162, (2021)
4. O. Malyuskin.: Microplastic Detection in Soil and Water Using Resonance Microwave Spectroscopy: A Feasibility Study. En in IEEE Sensors Journal, vol. 20, no. 24, pp. 1481714826, (2020)
5. NauticExpo, https://www.nauticexpo.es/fabricante-barco/vehiculo-no-tripulado-marino23029.html
6. Alexander T. Demetillo, Evelyn B. Taboada.: Real-Time Water Quality Monitoring For Small Aquatic Area Using Unmanned Surface Vehicle. En Engineering, Technology & Applied Science Research Vol. 9, No. 2, pp. 3959-3964 (2019)
7. Wonse Jo, Yuta Hoashi, Lizbeth Leonor Paredes Aguilar, Mauricio Postigo-Malaga, José M. Garcia-Bravo, Byung-Cheol Min.: A low-cost and small USV platform for water quality monitoring. En HardwareX, Volume 6, (2019)
8. G. Ferri *et al*.: Design, fabrication and first sea trials of a small-sized autonomous catamaran for heavy metals monitoring in coastal waters: En 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, pp. 2406-2411, (2011)
9. G. Ferri, A. Manzi, F. Fornai, F. Ciuchi and C. Laschi.: The HydroNet ASV, a Small-Sized Autonomous Catamaran for Real-Time Monitoring of Water Quality: From Design to Missions at Sea. En IEEE Journal of Oceanic Engineering, vol. 40, no. 3, pp. 710-726, (2015)
10. Difference between Heading (HDG) and Course Over Ground (COG), https://help.fleetmon.com/en/articles/6589334-difference-between-heading-hdg-and-courseover-ground-cog

11. NMEA-0183 Sentences for GPS Receivers, https://w3.cs.jmu.edu/bernstdh/web/common/help/nmea-sentences.php
12. Honeywell – Compass Heading Using Magnetometers, https://aerospace.honeywell.com/content/dam/aerobt/en/documents/learn/products/sensors /a pplication-notes/AN203_Compass_Heading_Using_Magnetometers.pdf
13. Leonardo Garberoglio, et al.: Diseño de un autopiloto para pequeños vehículos no tripulados. En Revista elektron, Vol. 3, No. 1, pp. 29-38, (2019)
14. Orange_Pi,http://www.orangepi.org/html/hardWare/computerAndMicrocontrollers/details/Orange-Pi-PC-2.html
15. Invensense, https://invensense.tdk.com/wp-content/uploads/2015/02/PS-MPU-9250A-01v1.1.pdf
16. Arbiam, https://www.armbian.com/
17. Linaro, https://www.linaro.org/about/

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Experiencia de desarrollo de una aplicación de reconocimiento de canciones mediante la técnica de huella de audio

Rodrigo Lago[1], Veronica Scholz[1], Roman Bond[1], Martin Morales[1,2], Diego Encinas[1,3]

[1] SimHPC-TICAPPS. Universidad Nacional Arturo Jauretche. Florencio Varela, 1888, Argentina.

[2] Centro CodApli. FRLP. Universidad Tecnológica Nacional. La Plata, 1900, Argentina.

[3] Instituto de Investigación en Informática (III-LIDI). Facultad de Informática, Universidad Nacional de La Plata - Centro Asociado CIC. La Plata, 1900, Argentina.

{rodrigazo98, vero.scholz}@gmail.com, {rbond, martin.morales, dencinas}@unaj.edu.ar

Abstract. El artículo presenta el desarrollo de una aplicación para reconocer canciones, utilizando la tecnología de Reconocimiento Automático de Contenido (ACR). Se comparan las soluciones ACR Cloud y Music Recognition para el procesamiento en la nube en tiempo real. También se realizan pruebas de rendimiento con diferentes calidades de audio y se analizan los resultados. Se discuten los usos de la huella digital de audio en la gestión de derechos de autor y el reconocimiento de canciones. La aplicación se muestra efectiva en reconocimiento de música en tiempo real, lo cual permite aplicar esta tecnología en otras áreas.

Keywords: audio, fingerprint, aplicación, Nyquist-Shannon, muestreo.

1. Introducción

El reconocimiento automático de contenido o ACR por sus siglas en inglés (Automated Content Recognition) se refiere a la capacidad de los sistemas informáticos para identificar y analizar de manera automática el contenido presente en medios como videos, imágenes y audio. En este trabajo se presenta el desarrollo de una aplicación que pueda reconocer canciones, una herramienta para docentes y estudiantes de conservatorios o escuelas de música. En un comienzo se decidió entre dos Clouds: AWS con la app Music Recognition [1] y ACRCloud. Debido a los cortos plazos de prueba de Music Recognition de Sensifai [2] y a la escasa documentación fue imposible ponerla a prueba y se decidió continuar el desarrollo de la aplicación con ACRCloud.

El desarrollo se divide en dos etapas. En la primera etapa, se definen las pruebas a realizar y qué metodología utilizar. En la segunda etapa se utilizó React Native y Expo para la implementación de la aplicación de reconocimiento de canciones. A su vez, se obtienen las métricas para su posterior análisis. Luego se incorpora un simple afinador que reconoce notas musicales.

2. Trabajos relacionados

Google en la actualidad permite buscar una canción diciendo “Hey Google que canción estoy escuchando”, pero en este caso aparece el video de la canción que lleva a YouTube. Devuelve la información relacionada y sobre el artista. Claro Música también cuenta con reconocimiento de canciones en este caso es pago junto a la suscripción. Todas estas aplicaciones utilizan ACR Cloud para el desarrollo del reconocimiento de sonido, esto se puede ver en la página de ACR Cloud donde se encuentra una lista de empresas que utilizan su servicio.

La plataforma KKBox [3] se ha convertido en una de las bibliotecas de música Asiática. Cuenta con más de 40 millones de pistas para elegir y está disponible únicamente en algunas regiones de Asia. Mediante una colaboración con ACR Cloud incluyeron en su plataforma un reconocimiento de música que permite a los usuarios la opción de “reconocimiento de música”.

Shazam [4] es una aplicación para móvil que permite la identificación de música. Permite grabar una muestra de música y, al igual que ACR Cloud, crea una huella digital acústica a partir de dicha muestra y se compara con una base de datos para encontrar coincidencias. En Shazam, se brinda al usuario la siguiente información:

- Nombre de la canción.
- Información de la banda: conciertos, otras canciones.
- Letra.
- Información de la pista.

3. Desarrollo

ACRCloud brinda servicios automáticos de reconocimiento de contenido mediante algoritmos de Huellas Dactilares de Audio y Consulta por Zumbido. Posee 40 millones de bases de datos de huellas digitales de música. Los usuarios gratuitos pueden enviar 100 consultas por día en un tiempo de respuesta de 2 a 4 segundos en promedio. Brinda un listado de librerías compatibles con React Native para hacer la conexión con su api (pero no para React js) [5].

Soluciones ACRCloud:

- Soporte de carga de contenido personalizado y reconocimiento.
- Monitoreo de transmisión en tiempo real para música o anuncios.
- Admite detección de canales de TV en vivo e interacción de segunda pantalla.
- Identificación de contenido con derechos de autor y servicio cumplimiento.
- Soporte de servicio de identificación fuera de línea para dispositivos inteligentes y aplicaciones.
- Medición de audiencia.

ACR Cloud puede hacer coincidir la "huella digital" o ID de una canción de destino (características acústicas clave como el tempo y los tonos de una pieza) con una base de datos de referencia de millones de pistas (Figura 1).

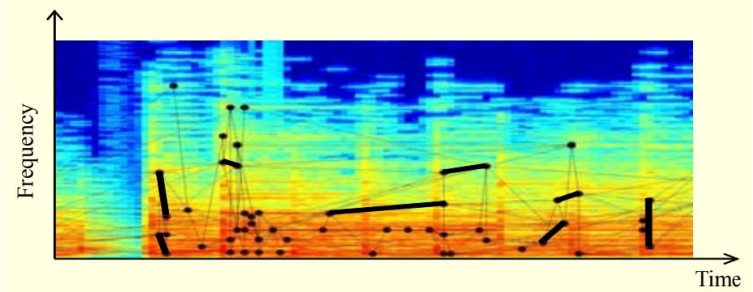


Fig. 1. Huella digital de audio [6].

El primer paso es la creación de la base de datos, donde se ingresan fragmentos de canciones, se extraen las huellas y se almacenan. En el siguiente paso, el usuario de la aplicación captura el audio, la app genera la huella y se envía la consulta a la base de datos. La respuesta indicará al usuario si hubo coincidencias. En la Figura 2 se puede observar los pasos a seguir.

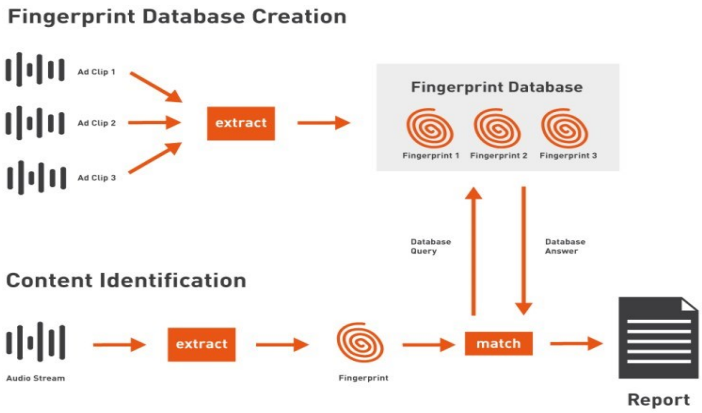


Fig. 2. Proceso de identificación de sonidos mediante huella digital. [7]

Entre los usos que se le pueden dar a la huella digital de audio se destacan:

- Derechos de autor: ACRCloud ayuda a controlar el uso de los derechos de autor de sellos discográficos. Las discográficas aplican los algoritmos de reconocimiento de contenido automatizado de la startup para monitorear obras de programas de radio y televisión, contenido generado por usuarios (UGC) en plataformas como YouTube o TikTok, o cualquier contenido que debería estar pagando derechos de autor.
- Reconocimiento de canciones: el reconocimiento de canciones de ACRCloud es una tecnología iniciada por Shazam, que los fabricantes de teléfonos (Huawei, Xiaomi) han integrado en sus dispositivos.

En [8] señalan que una huella digital de audio es una forma compacta basada en contenido que resume una grabación de audio.

La gráfica en el dominio del tiempo muestra los cambios de la amplitud de la señal con respecto al tiempo (Fig. 3.). Es una gráfica de la amplitud en función del tiempo, donde la fase y la frecuencia no se miden explícitamente. Un gráfico frecuencial muestra las componentes de la señal según la frecuencia en la que oscilan dentro de un rango determinado. Esta gráfica se tiene en cuenta para crear la huella digital.

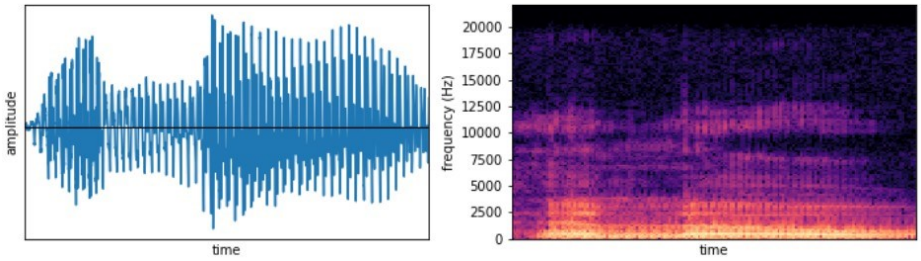


Fig. 3. Gráficas de amplitud en función del tiempo [5].

3.1 Primera etapa

Durante esta etapa se agregó React Native a la aplicación ya que, además de tener un buen desempeño en Mobile, permitió utilizar las librerías brindadas por ACRCloud. En conjunto con React Native, se incluyó Expo JS, que es un framework para poder trabajar con React Native el cual permite la emulación de dispositivos y la instalación de la aplicación en dispositivos físicos, lo que facilita el desarrollo.

3.2 Pruebas de rendimiento

Durante las pruebas de rendimiento se utilizó Python. Se trabajó con archivos locales de diferentes calidades para obtener los tiempos de respuesta favorables del cloud en función de las calidades de archivos enviados.

Se calcula el tiempo promedio que tarda en procesar archivos según la calidad del archivo para estimar la calidad de audio mínima y así obtener una respuesta en el menor tiempo posible. Entre las métricas que servirán de ayuda para obtener conclusiones se encuentran:

1. Tasa de error de detección: Esta métrica mide la cantidad de veces que el cloud no puede identificar correctamente una canción en función de la calidad del archivo.
2. Tiempo de procesamiento: El tiempo que tarda el algoritmo en procesar el archivo de audio en milisegundos.
3. Calidad de audio (baja, media, alta) depende del formato de audio y la frecuencia de muestreo en Hz.
4. Tamaño de archivo en bytes.
5. Duración del archivo: La duración del archivo de audio en segundos.

Durante estos cálculos surgieron los siguientes interrogantes: ¿La calidad del audio afecta significativamente la precisión de la detección de canciones? ¿El tiempo de procesamiento de un archivo de audio está correlacionado con su calidad?

3.2.1 Objetivo de las pruebas:

Medir los tiempos (en segundos) que tarda en procesar archivos de audio, el cloud de detección de canciones, según la calidad del archivo. Con el objetivo de estimar la calidad de audio mínima requerida para obtener una respuesta favorable y en el menor tiempo posible.

3.2.2 Metodología empleada:

Para construir el indicador se seleccionaron 4 calidades de audio (excelente, buena, regular y baja) por cada canción y se cargan carpetas por fragmento de canciones de 7 segundos. Se hace la solicitud y se reportan los resultados de *éxito* o *error.* También se obtiene el *coste de tiempo de procesamiento* de los casos favorables y se contabilizan los casos no favorables.

3.2.3 Fuente de información y tipos de registros:

Se envían archivos de audio (wav, mp3, ogg), luego se almacena el reporte (en formato json) y se analizan los resultados en un gráfico comparativo.

3.2.4 Tiempos esperados:

Se clasifica el tiempo de procesamiento a esperar en 3 categorías:

- Óptimo: Un tiempo de procesamiento de hasta 0.5 segundos
- Aceptable: Un tiempo de procesamiento entre 0.5 y 2 segundos
- No aceptable: Timeout, sucede cuando no se reconoce la canción.

3.2.5 Calidades empleadas:

Grupo 1: Excelente calidad

- Formato WAV (frecuencia de muestreo de 96000 KHz - 96 KHz, 2 canales).
- Formato FLAC (frecuencia de muestreo de 44100 KHz, 2 canales).

Grupo 2: Buena calidad

- WAV (frecuencia de muestreo de 48000 KHz - 48 KHz, 2 canales).
- MP3 (tasa de bits de 256Kbps, 2 canales).

Grupo 3: Calidad regular

- WAV (frecuencia de muestreo de 22050 KHz - 20 KHz, 2 canales).
- MP3 (tasa de bits de 128Kbps, 2 canales).
- FLAC (frecuencia de muestreo: 8000 KHz, 11025 KHz, 12000 KHz, 16000 KHz, 22050 KHz, 24000 KHz, 32000 KHz).

Grupo 4: Baja calidad

- MP3 (tasa de bits de 64 Kbps, 1 canal).
- AMR (tasas de bits constantes: 12.20 Kbps, 10.20 Kbps, 7.95 Kbps, 7.40 Kbps, 6.70 Kbps, 5.90 Kbps, 5.15 Kbps, 4.75 Kbps)

3.2.6 Resultados obtenidos

Entre los resultados de la medición se observó que algunos archivos de mayor calidad consumen tiempos de procesamiento más largos. Esto se debe, en primer lugar, a que algunos archivos de mayor calidad tienden a ser más grandes en tamaño, lo que conlleva un tiempo adicional de carga, como se puede apreciar en la Figura 4. A su vez, estos archivos pueden contener mayor cantidad de información, aumentando la complejidad del procesamiento y originando tiempos de detección más extensos en comparación con archivos de menor calidad.

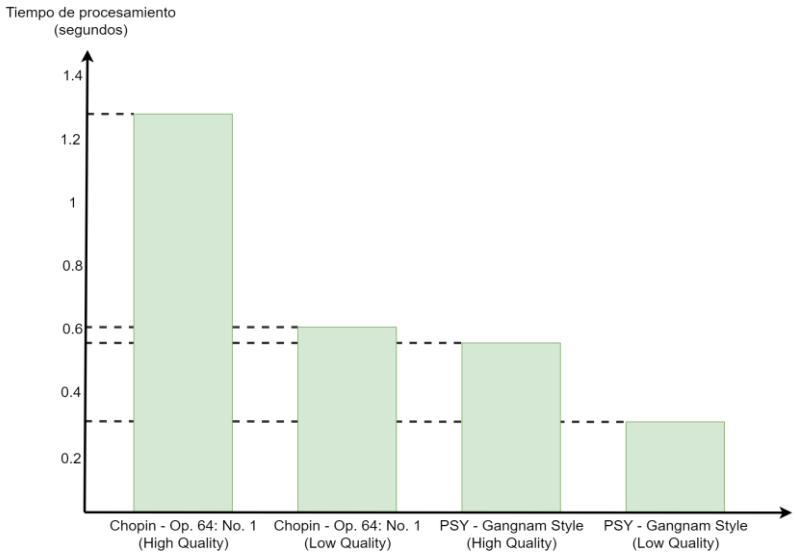


Fig. 4. Gráfico de tiempos de procesamiento de pruebas realizadas

Sin embargo, se puede observar que en casos donde las grabaciones de las canciones poseen más cantidad de ruido y menos definición en los instrumentos, como en Sodom - M 16 (ver Figura 5), al reducir la calidad del archivo, no se realiza el match con la canción original.

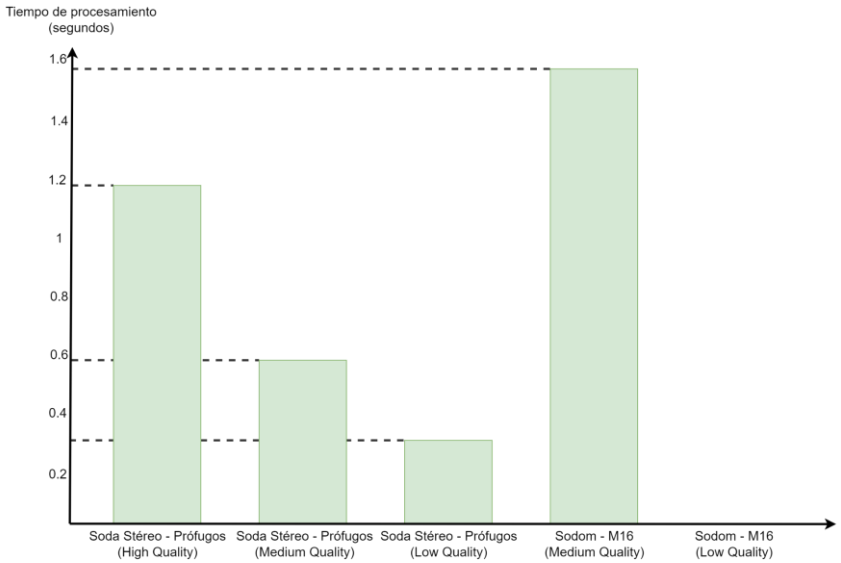


Fig. 5. Gráfico de tiempos de procesamiento de pruebas realizadas.

3.3 Muestreo del audio

El sonido es una vibración que se propaga como una onda mecánica de presión y se desplaza en el medio (ej: aire). El micrófono traduce esta señal de presión continua en una tensión analógica continua [9]. Luego se realiza una conversión analógico-digital. Como siempre se introduce una pequeña cantidad de errores al cuantificar la entrada, se realizan muchas conversiones en pedazos muy pequeños de la señal, el "sampling".

Según el teorema de Nyquist-Shannon [10] la tasa de muestreo necesaria para capturar determinada frecuencia debe ser el doble de la frecuencia muestreada, es decir, para capturar todas las frecuencias del oído humano (20 Hz-20 KHz) se debe realizar el el muestreo a 44.100 Hz.

Como resultado del sampling obtenemos una señal grabada en función del tiempo, el cambio de amplitud de la señal a lo largo del tiempo. A través de Fourier sabemos que una señal puede descomponerse en varias señales sinusoidales simples, cuya sumatoria es igual a la señal descompuesta. La serie de los sinusoides que juntos forman la señal en el dominio del tiempo original se conocen como serie de Fourier. La transformada de Fourier deconstruye una representación en el dominio de tiempo de una señal en la representación del dominio de frecuencia.

Trabajar con funciones en términos de la frecuencia facilita el análisis de la señal digital para estudiar el espectro y determinar las frecuencias que están presentes y/o faltan, hacer filtrados, aumentar frecuencias o reconocer el tono exacto de las frecuencias.

En el caso del afinador, para calcular la frecuencia en tiempo real, se tiene un bucle de grabación (Figura 6) donde se almacenan muestras de audio en el buffer, luego se procesan las muestras de audio y finalmente, se vacía el buffer.

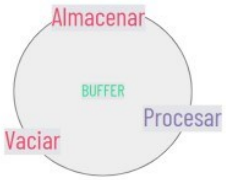


Fig. 6. Bucle de grabación.

El algoritmo recibe la muestra de audio y la multiplica por una función ventana para evitar bordes y aplica la transformada rápida de fourier para obtener la señal en su representación en el dominio de la frecuencia.

Es la misma representación estática que, al igual que en la huella de audio digital, actúa como un tipo de firma dactilar de la señal.

3.4 Segunda etapa

Luego de esta primera fase se utilizó el framework de front-end React Native para el desarrollo móvil e híbrido además de Expo JS para la emulación y pruebas en los distintos dispositivos. Utilizados ambos en conjunto brindan un ecosistema bastante completo para el desarrollo y pruebas de una aplicación móvil híbrida.

Una vez implementado el código, ACR Cloud proporciona librerías las cuales son la configuración para obtener el reconocimiento de lista de canciones. Estas librerías brindan las herramientas para consultar en la nube y traer la información deseada.

ACRCloud facilita que las consultas sean a la nube o de modo offline pero el objetivo es ver la funcionalidad de la nube. En el caso de querer implementar un sistema de reconocimiento de canciones en modo offline, todo el procesamiento se realiza en el dispositivo móvil.

3.5 Cálculos

Para el proyecto se evaluó distintas calidad de reproducción y el tiempo de respuesta por cada tema se tomó 50 muestras con variación de calidad. Para hacer el reconocimiento, se hizo en ambiente libre de sonidos externos, con variación de partes de la canción este programa toma 10 segundos en cada muestra. Resultó útil realizar la probabilidad que se encuentre cierta canción variando sus calidades de reproducción, de estos casos favorables se realizó un promedio de respuesta del software lo cual se aprecia en las Figuras 4 y 5 . La siguiente ecuación se utiliza para saber la probabilidad de encontrar con éxito una canción. P(A) es la probabilidad de que encuentre la canción, casos favorables son las veces que se encontró la canción y los casos posibles son la cantidad de casos analizados.

$$P(A) = \frac{Casos\ Favorables}{Casos\ Posibles} \quad (1)$$

De los aciertos se realizó el promedio para representar en los gráficos por lo que:

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \quad (2)$$

Donde X es el promedio. En este caso la canción es poco probable de encontrarla en baja calidad, pero el tiempo de respuesta es menor al de mayor.

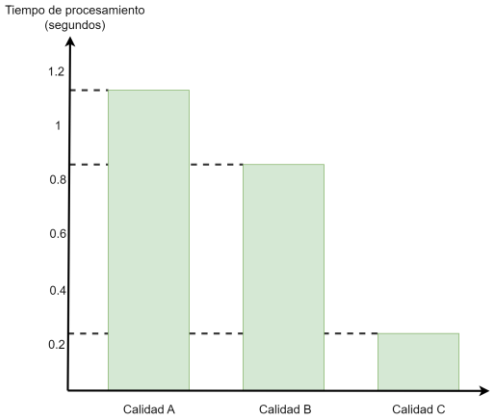


Fig. 7. Gráfico comparativo de calidades.

En la Figura 7 se analizan tres calidades de sonido. En la barra A se obtuvo un promedio de 1,17 de los casos favorables que fueron un total de 28 de 50 pruebas por la ecuación 1.1 la probabilidad de encontrarla es 0,56. La barra B se trata de una calidad media, con un total de 43 casos favorables de 50, y su probabilidad es de 0,86. En la barra C el tiempo de respuesta promedio es de 0,24, los casos favorables fueron 30 lo que da 0,6 en este caso la probabilidad de encontrar el tema tuvo mayor éxito con la producción regular y baja.

Luego de iniciar la aplicación aparecerá un botón que dice buscar, se pasa a detallar los pasos que sigue el algoritmo implementado.

1. Primero se pide el permiso para poder acceder al micrófono.
2. Se debe presionar el botón empieza a grabar durante 7 segundos la canción.
3. La grabación se envía al servidor de ACRCloud.
4. Si encuentra la canción devuelve un archivo JSON con varios datos.
5. Luego se renderiza el nombre de artista, canción y tiempo de procesamiento del cloud.
6. Si no la encuentra devuelve null y en pantalla se ve “canción no encontrada”.

4. Conclusión y trabajos futuros

Se puede observar que al desarrollar una aplicación en tiempo real los tiempos de respuestas son mínimos con respecto a otros. En cuanto a los cálculos se vio una variación en los aciertos a la calidad y esto se puede asociar a que el tema analizado

pudo tener mayor reproducción que otros. Con este tipo de aplicación se puede ayudar a un grupo de personas, de una determinada actividad, pero esta aplicación sólo podría ser un ejemplo para futuros desarrolladores que quieran facilitar en un aspecto la vida de otros usuarios. Considerando que un sistema similar podría reconocer diferentes tipos de ruidos, se puede tomar como ejemplo la detección de ruidos en un motor o mecanismo. Al realizar pequeñas modificaciones al código actual, se podrían resolver problemas adicionales para los usuarios, abordando diversas problemáticas. De esta manera, se mejoraría la funcionalidad del sistema, permitiendo su aplicabilidad en distintos contextos y maximizando su utilidad para beneficio general.

Referencias

1. https://aws.amazon.com/marketplace/pp/prodview-tnt5od5jzxct2
2. https://sensifai.com/
3. Zheng, Hanyue. KKBox subscription prediction: an application of machine learning methods. Diss. 2018.
4. Wang, Avery. "The Shazam music recognition service." Communications of the ACM 49.8
5. (2006).
6. https://docs.acrcloud.com/
7. Kim, Hyoung-Gook, and Jin Young Kim. "Robust audio fingerprinting method using prominent peak pair based on modulated complex lapped transform." ETRI Journal 36.6 (2014).
8. https://www.mufin.com/company/technology/
9. Cano, Pedro, et al. "Audio fingerprinting: concepts and applications." Computational intelligence for modelling and prediction (2005).
10. Oliver Gil, José Salvador. "Captura de señales de audio y preparación para su compresión." (2018).
11. Por, Emiel, Maaike van Kooten, and Vanja Sarkovic. "Nyquist–Shannon sampling theorem." Leiden University 1.1 (2019).

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Sistema de Adquisición de Datos Utilizando Sensores de Baja Complejidad para Tractores Antiguos del Alto Valle de Río Negro y Neuquén

Rafael Ignacio Zurita, Naiara Sheffield, Marcelo Moreyra,

Dario Mendieta, Favio Masson, and Miriam Lechner

Depto. Ing. de Computadoras, Facultad de Informática
Depto. de Electrotecnia, Facultad de Ingeniería
Universidad Nacional del Comahue. Buenos Aires 1400, Neuquén, Argentina.
{rafa,naiara.sheffield}@fi.uncoma.edu.ar,
{marcelo.moreyra,dario.mendieta}@fain.uncoma.edu.ar,
fmasson@uns.edu.ar,mtl@fi.uncoma.edu.ar
http://www.fi.uncoma.edu.ar - http://www.fainweb.uncoma.edu.ar

Resumen La navegación autónoma en fruticultura esta´ en investigación y desarrollo principalmente en países desarrollados. Las soluciones actuales involucran computadoras de alto rendimiento y sensores específicos de alta complejidad. También se requiere adaptar las chacras al tractorrobot autónomo, lo que implica grandes inversiones. Sin embargo, regionalmente, el 74% de las chacras frutícolas del Alto Valle de Río Negro y Neuquén pertenecen a pequeños productores familiares, quienes cuentan en promedio con un tractor antiguo, cuyo modelo varía entre 1970 a 2000. Con el fin de modelar a largo plazo una solución de navegación autónoma en fruticultura local se presenta, en este artículo, la arquitectura de un sistema embebido, que utiliza sensores de baja complejidad, para capturar las señales requeridas en el proceso de posicionamiento (localización y orientación) de cualquier tractor antiguo regional en su entorno de trabajo. Se presenta también una evaluación de la frecuencia de muestreo alcanzado por los sensores seleccionados y el software desarrollado, y una evaluación de la señal de video que es posible capturar con el sistema planteado, utilizando un algoritmo del estado del arte de Visual-SLAM.

Keywords: Navegación Autónoma, Fruticultura, Adquisición de Señales, Sistema Embebido de Tiempo Real, Sensores de Baja Complejidad.

1. Introducción

En el Norte de la Patagonia argentina, en el “Alto Valle” de Río Negro y Neuquén, se desarrolla la cadena productiva de manzana y pera. En el año 2020 el total de la superficie plantada con manzanos y perales alcanzaba las 35.609 ha, y el 74% de los establecimientos frutícolas (aproximadamente 1200) son explotados por pequeños productores familiares [1,2]. En estas explotaciones tradicionales

el productor y su familia realizan casi la totalidad de las acciones de manejo cultural de la chacra, contratando mano de obra transitoria durante la cosecha, el raleo o la poda.

En la actualidad la maquinaria agrícola más utilizada en el Alto Valle para facilitar las tareas rurales es el tractor (viñatero, frutero y/o convencional); y está presente en

todas las tareas de la actividad fruti-hortícola: desde la preparación del suelo, el desmalezado, la protección de los cultivos y la cosecha hasta el transporte de la producción [4]. Cada pequeño productor suele ser propietario de un tractor (dos a lo sumo), y los modelos de los mismos varían desde el año 1970 al año 2000 [5].

Durante las últimas dos décadas se han logrado adelantos en el desarrollo de vehículos autónomos agrícolas [6,7]. Un sistema de conducción autónomo puede controlar un vehículo y navegar en su ambiente sin necesidad de intervención humana [8]. Un vehículo equipado con este sistema percibe el ambiente, localiza su posición, y opera el vehículo para lograr las trayectorias que le permitan alcanzar las rutas señaladas. Existen tractores con autopilotos desde el año 2000, que fueron diseñados para trabajar en extensiones de campo abierto, sobre todo en los de producción cerealera. Un ejemplo comercial es el autopiloto AutoTrac, de John Deere [9]. En fruticultura, las cortinas de álamos y las alturas de las hileras de árboles, en montes frutales como los del Alto Valle, complican obtener una localización con la precisión necesaria para los sistemas de autopiloto, debido a que las señales de los satélites no arriban con visión directa a los GPS [10]. En los últimos años se han desarrollado sistemas de autopiloto con técnicas alternativas para fruticultura en países desarrollados. Sin embargo, estos desarrollos se enfocan en sensores específicos de alto costo, computadoras de alto rendimiento, y técnicas que requieren adecuar el monte frutal a la máquina, ya sea porque este es nuevo y se lo diseña en base a las capacidades del autopiloto, o porque ya existe pero debe ser probablemente reconvertido, lo que implica grandes inversiones.

En este trabajo se presenta una nueva aproximación que pretende tener desempeños adecuados a la aplicación, pero con restricciones de cómputo y sensado que haga viable su implementación para pequeños productores frutihortícolas. La arquitectura propuesta utiliza una computadora embebida y sensores de baja complejidad para capturar las señales vitales requeridas en el proceso de posicionamiento (localización y orientación) de cualquier tractor antiguo regional en su entorno de trabajo. Se presenta también una evaluación de la frecuencia de muestreo alcanzado por los sensores seleccionados y el software desarrollado, y una evaluación de la señal de video que es posible capturar con el sistema planteado, utilizando un algoritmo del estado del arte de Visual-SLAM. Los resultados muestran que el sistema propuesto exhibe características iniciales adecuadas para el desarrollo de un sistema de posicionamiento adaptable a la maquinaria existente de pequeños productores, y que pueda ser integrado, a futuro, en la implementación de un sistema mayor de navegación autónoma en fruticultura regional.

El resto del presente artículo está estructurado como sigue: en el siguiente Capítulo se describen los trabajos relacionados anteriores. En el Capítulo 3 se presenta la arquitectura del sistema propuesto, y una guía de pautas de cómo seleccionar la computadora embebida. En el Capítulo 4 se realiza una evaluación de la frecuencia alcanzada por los sensores, y una validación de los datos de vídeo adquiridos por el sistema. Finalmente, se detalla, en el Capítulo 5, conclusiones y trabajos futuros de esta investigación en progreso.

2. Antecedentes

En general, las técnicas en desarrollo para navegación autónoma en montes frutales se enfocan en tres ejes principales: la modificación o formación de monte frutal nuevo para adecuarlo al tractor-autónomo, la utilización de cámaras estéreos en conjunto con computadoras de alto rendimiento para la ejecución de algoritmos de Visual-Slam, y la utilización de escáneres láser como sensor de percepción local.

Diversas publicaciones han tratado este problema con algunos de estos enfoques. En [11] por ejemplo, se utiliza un escáner láser 2D para guiar un tractor a lo largo de una calle entre hileras de frutales. En [12] se agrega, además, cintas reflectivas en cada extremo de la hilera, con el fin de detectar con el escáner láser el final de la misma. En [13] se emplean múltiples sensores, incluyendo un escáner láser, una cámara de video estéreo y una unidad de medición inercial (IMU) para guiar a un vehículo en una plantación de naranjos. Además se logra un error lateral promedio de 0,05m a una velocidad de 3,1m/s, y se logra girar al finalizar la calle para continuar en la siguiente hilera. En [14] se describe una plataforma automatizada guiada por dos escáneres láser 2D, que ha recorrido más de 300 km de calles de frutales. En [15] se presenta un sistema de varios tractores para desmalezar y fumigar una plantación de naranjos. Cada tractor autónomo utiliza un escáner láser 2D y una cámara estéreo a color para percibir el ambiente, y ha conducido al menos 1000 km. En [16] se detalla el desarrollo de un robot autónomo capaz de transportar un bin vacío y reemplazarlo por uno lleno en la plantación. Utiliza un GPS para guiar al vehículo fuera de las hileras del monte y dos escáneres láser 2D y una IMU para cuando trabaja dentro de las hileras. El robot posee cuatro ruedas de giro y dirección independientes, lo que permite realizar maniobras de giro omnidireccionales.

Cabe agregar, a lo mencionado al principio de esta sección, que un escáner láser, como los utilizados en los antecedentes presentados, tiene un costo aproximado que supera los USD 3500, y su precio no ha disminuido notablemente en los últimos diez años. Tampoco se tuvieron en cuenta a las cortinas de álamos encontradas comúnmente como protección al viento en las chacras del Alto Valle, debido principalmente a que estos desarrollos se implementan en plantaciones nuevas que no lo requieren. A diferencia de los trabajos previos el objetivo de nuestro trabajo está puesto en utilizar herramientas, técnicas y métodos asequibles a los pequeños productores familiares, y que puedan aplicarse a los tractores y montes frutales ya presentes en Alto Valle.

3. Arquitectura y Desarrollo del Sistema de Adquisición

3.1 Arquitectura del hardware

En la Figura 1 (izquierda) se puede observar un diagrama de bloques de alto nivel de la arquitectura del hardware. Específicamente, el hardware está compuesto en su parte central por una computadora embebida Banana Pi M2-Ultra. Entre sus características más importantes encontramos un SOC (System On Chip) que contiene un microprocesador con 4 CPUs ARM, de 1,2Ghz, 2GB de RAM, 8GB de disco Flash, una interfaz para discos SATA, 3 puertos USB, radio WiFi, dos buses i2c, y puertos GPIOS. Además, se seleccionaron los siguientes sensores de baja complejidad: un magnetómetro QMC5883L, un giróscopo ITG3200, un acelerómetro ADXL345, un

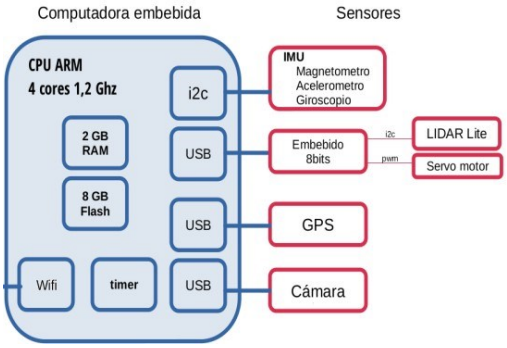
lidar lite v4, una cámara de vídeo monocular, y un GPS u-blox evk-m8. Como puede observarse en la Figura 1 (derecha) los sensores junto con la computadora embebida son de bajo costo, y su valor es de un orden de magnitud menor a los autopilotos propuestos en la literatura. Los sensores adquiridos en conjunto con la computadora central embebida conforman el *hardware del sistema de adquisición de datos*.

Pautas para la elección de la computadora embebida

El proceso de selección y adquisición requirió un análisis comparativo entre varias opciones existentes en el mercado. En particular, la computadora embebida es el componente más crítico, porque todos los datos son adquiridos, pre-procesados, almacenados y enviados desde este sistema. Con el fin de cumplir el requisito de que el sistema sea asequible, pero a su vez tenga las características apropiadas, se determinaron una serie de pautas para su selección.

Enumeramos a continuación las pautas definidas, que pueden ser de aplicación en futuros trabajos donde se necesite realizar una selección similar:

1. **Soporte del kernel Linux mainline**. Es importante que el proyecto del kernel Linux oficial tenga soporte para el SOC. Muchas empresas proveen únicamente un kernel Linux propio. El riesgo aquí es que las nuevas versiones del kernel oficial, que corrijan errores, o provean nuevas funcionalidades, no podrán usarse a futuro. Además, un kernel funciona, generalmente, con un tipo de distribución o versiones de bibliotecas, etc. Si se desea utilizar una distribución Linux más reciente o nuevas versiones de cierto software o bibliotecas, puede llegar a no ser posible con un kernel Linux del fabricante. Este riesgo no es menor, porque una gran cantidad de computadoras embebidas del mercado ofrecen únicamente su propio Linux.



HW tipo	Marca/Modelo	
Computadora Embebida	Banana Pi M2-Ultra	65
Acelerómetro	Adxl345	15
Giróscopo	ITG3200	15
Magnetómetro	QMC5883L	10
Cámara monocular	Monocular HD Cam	50
Lite lidar	Garmin LIDAR-Lite v4	65
GPS	U-blox evk-m8	180
Costo total aproximado (USD)		**335**

Base del hw utilizado en técnicas actuales (para comparativa)	
Computadora Embebida Nvidia Jetson tx2	800
Lidar 2D SICK	3500
Camara estéreo	789
Costo total aproximado (USD)	**4289**

Figura1: Izquierda: diagrama de bloques del hardware del sistema de adquisición. Derecha: listado de componentes seleccionados y comparativa de precio del hardware.

2. **Determinar anchos de banda del hardware desde las hojas de datos**. No sólo se deben observar las características generales del hardware, se debe también verificar los requisitos en sus hojas de datos o esquemáticos. Por ejemplo, la placa Raspberry Pi 3 ofrece varios puertos USB 2.0. Sin embargo, el controlador ethernet en esa Raspberry Pi, es externo al SOC, y está conectado al hub USB del sistema. Por lo tanto, gran parte del ancho de banda USB estará siendo utilizado por el controlador ethernet si se usa

esta interfaz, y el restante podría no ser suficiente para el sensor conectado a otro puerto. Estas especificidades sólo son detalladas en tales documentos como las hojas de datos y los esquemáticos.

3. **Fabricante de familias de computadoras embebidas**. La placa seleccionada será probablemente obsoleta en algún tiempo, entonces discontinuada, y no más disponible en el mercado. Si el proyecto de investigación tiene líneas de trabajos extensas conviene entonces seleccionar una placa de un fabricante que desarrolle cada nueva versión en base a un SOC de la misma familia. Esto permite migrar el trabajo a la nueva placa de manera sencilla. Se requiere observar la línea de tiempo del fabricante y cómo trabaja cada nuevo producto.

3.2 Arquitectura del software

El software de adquisición de datos está compuesto de piezas particulares de software que desarrollamos para la adquisición de datos de cada sensor. Un diagrama de bloques en niveles del software puede observarse en la Figura 2. Los bloques en rojo son los desarrollados en este trabajo. En particular, el software de adquisición de imágenes de la video cámara, de GPS y del lidar-lite fueron desarrollados en C, utilizando únicamente la biblioteca estándar de C. El acceso directo a los datos de cada sensor desde C y el driver del kernel Linux nos posibilita realizar una frecuencia de adquisición controlada. Por ejemplo, es posible adquirir a la mayor frecuencia posible si fuese necesario, sin intermediar otras capas de software, filtros, o buffers. [1]

Marcas de tiempo como dato central de interrelación

Si bien en principio los componentes del software son independientes, cada componente le incorpora, a cada muestra adquirida de su sensor específico, una marca de tiempo con resolución de microsegundos, obtenida desde un mismo reloj (reloj del kernel Linux). De esta manera, todos los datos adquiridos están inter-relacionados en una misma línea temporal. Por ejemplo, de cada foto se podría indicar a qué ubicación de GPS pertenece, o con qué dato de lidar o giróscopo. Incluso la relación puede ser de uno a muchos, ya que una foto estaría relacionada por ejemplo a un conjunto de datos cercanos en tiempo del giróscopo, pues la frecuencia de muestras del giróscopo es mucho mayor a la de la cámara. Todo esto es fundamental al momento de utilizar las mediciones en un esquema de fusión de sensores, típico en robótica para la resolución del problema de navegación.

[1] Las piezas de software desarrolladas para este sistema de adquisición se distribuyen bajo una licencia de software libre http://github.com/zrafa/ryva.

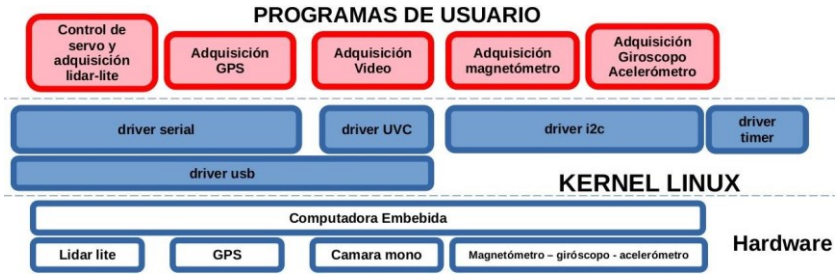


Figura2: Arquitectura del software. Los bloques en rojo son los desarrollados.

4. Evaluación

La evaluación del sistema se realizó mediante una experimentación en campo, adquiriendo los datos con la mayor frecuencia posible por este sistema, y procesando luego la información de vídeo utilizando un algoritmo de Visual-SLAM.

Para esto, se empotró el sistema en un tractor Deutz del Alto Valle, modelo 1970. En una caja estanca se colocó la computadora embebida y algunos de los sensores. Se construyó una caja contenedora exterior en un sector disponible del tractor antiguo y se colocó ahí la caja estanca. La cámara se ubicó en el frente del tractor, y el GPS sobre la caja del sistema; como puede verse en la Figura 3 (derecha). Luego de calibrar el sistema se comenzó la adquisición de datos de todos los sensores. Un conductor entonces realizó una navegación con el tractor realizando un circuito cerrado, a través de dos hileras de frutales de 175 metros cada una. Hileras que pertenecen a la chacra 28, de la calle A15, Cipolletti (RN). En la Figura 3 (izquierda) puede observarse en rojo las dos hileras recorridas.

4.1 Evaluación de las frecuencias de los sensores

El objetivo de este análisis es conocer si la frecuencia alcanzada es adecuada para una futura navegación autónoma. Como referencia, se recabaron las frecuencias alcanzadas en publicaciones previas sobre la temática [17,13]. En el Cuadro 1 puede observarse la frecuencia obtenida por el sistema para cada sensor, y en la siguiente columna, un promedio de las frecuencias establecidas en otras publicaciones de la temática. Los resultados muestran que las frecuencias están por encima de las establecidas en otros trabajos (a excepción de GPS comparado contra un GPS rtk). Cabe remarcar, que en versiones del software preliminares, en donde se reutilizó software ya existente, se obtuvieron frecuencias menores para algunos sensores. Esas evaluaciones anteriores fueron la motivación por la cual desarrollamos nuestras versiones de software de adquisición.



Figura3: Izquierda: Vista del recorrido. Derecha: Tractor con la caja adquisidora.

Cuadro 1: Comparación de Frecuencias en adquisición de datos

— Sensor	Frecuencia obtenida (hz)	Valor de Referencia (hz) —
Magnetómetro	30	-sin valores-
Lidar Lite	6 (obstáculo a 5mts)	3 (driver oficial, obstáculo a 5mts)
Giróscopo	153	140
Cámara	15	15
Acelerómetro	153	140
GPS	1	5 (con rtk)

4.2 Evaluación de los datos obtenidos por la cámara

Durante el recorrido del tractor se obtuvieron 9201 imágenes de 672x376 píxeles cada una. La evaluación de estos datos, utilizando un algoritmo de Visual-SLAM, requirió una serie de pasos sistemáticos:

1. Calibrar la cámara.
2. Realizar la adquisición de datos.
3. Procesar los datos con el algoritmo de Visual-SLAM.
4. Si se requiere, sintonizar el algoritmo, y volver al paso 3.
5. Evaluar la salida del algoritmo contra una medición ground truth [2].
6. Iterar el paso 3., 4. y 5. hasta lograr una sintonización del sistema consistente(o determinar que no es posible reconstruir el mapa y la trayectoria en este ambiente y/o con este sistema).

Algoritmo de Visual-SLAM

Los algoritmos de Visual-SLAM (Visual Simultaneous Localization And Mapping) utilizan solamente información proveniente de sensores visuales (cámaras) para resolver el problema de localización del tractor y simultáneamente la reconstrucción

[2] ground truth es el valor real o más cercano al real con el cual se debe comparar, generalmente obtenida desde una medición con gran precisión y exactitud.

del mapa del ambiente que transita. Para este trabajo, se ha optado por utilizar CCM-SLAM (Centralized Colaborative Monocular Simultaneous Localization And Mapping) [18]. Esta implementación es open source y se encuentra dentro de la categoría de los algoritmos que sólo utilizan imágenes de cámaras monocular.

En términos generales, la base para todos los algoritmos de Visual-SLAM es detectar entre dos imágenes consecutivas las mismas marcas naturales de ese ambiente. Estas características distinguibles o puntos 2D pueden ser diferencias de color, bordes o esquinas. La información de cómo mínimo 5 pares de marcas o puntos 2D que se correspondan entre dos frames, permite estimar la posición de la cámara y la ubicación 3D de estos puntos 2D. Este procedimiento se repite frame a frame, resultando en obtención de la trayectoria del vehículo y construcción del mapa 3D.

Configuración, Sintonización y Ejecución de Visual-SLAM

Los algoritmos de Visual-SLAM deben ser configurados y sintonizados. Durante la configuración se especifica la cantidad de fotos por segundo, la resolución de las imágenes, y los parámetros intrínsecos de la cámara. Una vez configurado, se continuó con un proceso sintonización del software. En el caso de CCM-SLAM, la sintonización se centra en la etapa de detección de características ORB(Oriented FAST and Rotated BRIEF - SLAM). La complejidad en este paso radica en que, en general, los algoritmos de Visual-SLAM requieren un ambiente e imágenes con ciertas condiciones:

1. Las imágenes deben estar enfocadas. Si esto no sucede (imagen borrosa) una característica detectada pueda no ser reconocida en las siguientes imágenes.
2. Ambientes no dinámicos. Si en el ambiente hay cambios constantes el algoritmo puede detectar falsos positivos que acarrean errores a todo el sistema.
3. Ambientes no regulares. Si existen patrones repetitivos, con texturas muy similares, los descriptores de las características de la imagen resultan similares entre sí, lo que puede acarrear falsos positivos de correspondencias.

Para el caso particular de las chacras del Alto Valle, las condiciones 1 y 3 no se cumplen completamente. El tractor vibra todo el tiempo, con lo cual es difícil que la cámara obtenga siempre fotos con buen enfoque. Además, las hileras de frutales son de patrones regulares, dado que el patrón del arbolado es repetitivo, y las texturas siempre similares. Es por este motivo que se considera a este ambiente particular como *hostil*. En nuestra experimentación en campo, el ambiente hostil demandó varias iteraciones del proceso de sintonización y ejecución, ajustando las variables ORBextractor* hasta que se logró una trayectoria que visualmente presentaba caracterísiticas similares a la obtenida de ground truth.

Validación de la trayectoria obtenida con ground truth

La trayectoria obtenida desde los datos de GPS es, en este trabajo, el ground truth, y se encuentra en un sistema de coordenadas global geodésico de la tierra. En cambio, la trayectoria determinada por CCM-SLAM se encuentra en un sistema de coordenadas que se desplaza con el vehículo. Además, al utilizarse una cámara monocular el sistema CCM-SLAM no determina la escala de distancias, debido a que no puede determinar la profundidad de la imagen a partir de una una única foto en cada momento. Es decir, no

existe una distancia en una unidad de longitud entre dos puntos de la trayectoria computada.

Con el fin de poder equiparar y comparar ambas trayectorias se utilizó el método de ajuste de escala y alineación Umeyama [19], a través del software *evo*[3]. El método Umeyama se utiliza para la alineación y registro de conjuntos de puntos en el espacio tridimensional. Su objetivo es encontrar la mejor transformación rígida (rotación, traslación y escala) entre dos conjuntos de puntos, minimizando la suma de las distancias cuadradas entre los puntos correspondientes en ambos conjuntos. La transformación rígida implica que los puntos conservan sus distancias y ángulos relativos después de aplicar la transformación.

Luego de aplicar el método se obtuvo como resultado la comparativa que puede observarse en la Figura 4. En trazado negro punteado se observa la trayectoria de la navegación realizada por el tractor (ground truth). La trayectoria en color azul fue la computada por CCM-SLAM. El resultado muestra una gran correlación entre ambas trayectorias, con un error determinado preliminarmente desde la gráfica de +/- 5 metros, lo que demuestra que es factible lograr implementar, con las restricciones impuestas, un algoritmo de localización complejo.

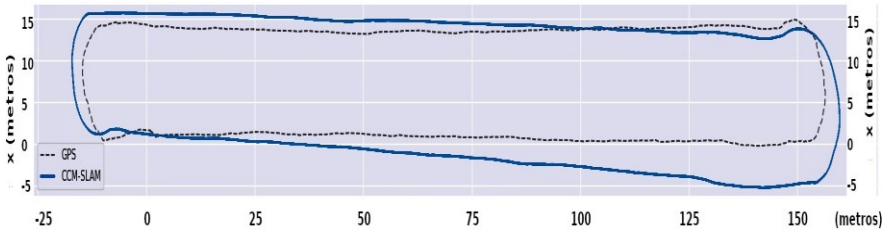


Figura4: Trayectoria obtenida con CCM-SLAM, comparada contra el *ground truth*.

5. Discusión / Conclusiones

En este trabajo se detalló la arquitectura del hardware y software de un sistema embebido para tractores antiguos, que adquiere datos de sensores de baja complejidad, para ser integrado, a futuro, en un sistema de navegación autónoma en fruticultura. Los resultados obtenidos muestran que las frecuencias de adquisición están dentro de los valores de referencia, y que los datos obtenidos en campo desde la vídeo cámara son convenientes para ser procesados por un sistema de posicionamiento futuro. Existen varias tareas a continuar en este trabajo. En particular, se requiere determinar la caracterización completa de cada sensor de baja complejidad en el ambiente frutal, la confección de un sistema de autocalibración (necesario en momentos previos a su uso en campo), y un modelo e implementación de integración y fusión de datos de los sensores.

[3] https://github.com/MichaelGrupp/evo

Referencias

1. Servicio Nacional de Sanidad y Calidad Agroalimentaria: Anuario estadístico 2020 - Centro Regional Patagonia Norte. Senasa, ISSN 2545-8124, 2019.
2. J. Toranzo.: Producción mundial de manzanas y peras. 1ra. ed. - Allen Río Negro: Ediciones INTA, ISBN: 978-987-521-696-9, 2016.
3. Smith, T.F., Waterman, M.S.: Identification of Common Molecular Subsequences.J. Mol. Biol. 147, 195–197 (1981)
4. A. Di Prinzio, C. Magdalena, S. Behmer.: El tractor en cultivos intensivos. Nocionesde uso y funcionamiento. 1a. ed. INTA, ISBN: 978-987-679-060-4, 2011.
5. ONU para la Alimentación y la Agricultura. Sujetos Sociales: Productores Agropecuarios, Pobl. Rural y Pueblos Originarios de la Prov. de Río Negro. 2015.
6. Q. Zhang, H. Qiu.: A Dynamic Path Search Algorithm for Tractor Automatic Navigation. Trans. ASAE 2004, 47, 639–646. doi 10.13031/2013.16027
7. T. Roberson, L. Jordan.: RTK GPS and automatic steering for peanut digging. Appl. Eng. Agric., 30, 405–409, 2014. doi 10.13031/aea.30.10432
8. R. Siegwart, I. R. Nourbakhsh, D. Scaramuzza.: Introduction to Autonomous MobileRobots. 2da. ed., The MIT Press, ISBN-13: 978-0262015356, 2011.
9. A. Marsh.: John Deere and the Birth of Precision Agriculture. IEEE Spectrum 2018.
10. V. Subramanian, F. Burks, A. Arroyo.: Development of machine vision and laser radar based autonomous vehicle guidance systems for citrus grove navigation. Comput. Electron. Agric., 53, 130–143, 2006. doi 10.1016/j.compag.2006.06.001
11. O. Barawid, A. Mizushima, K. Ishii, N. Noguchi.: Development of an Autonomous Navigation System using a Two-dimensional Laser Scanner in an Orchard Application. Biosyst. Eng., 96, 139–149, 2007. doi 10.1016/j.biosystemseng.2006.10.012
12. M. Bergerman, S. M. Maeta, et al: Autonomous Orchard Vehicles Help Tree Fruit Production. IEEE Robotics & Automation Magazine, vol. 22, no. 1, pp. 54-63, 2015.
13. V. Subramania, T. Burks.: Autonomous vehicle turning in the headlands of citrus groves. ASAE Annual Meeting, MN, USA. 2007. doi 10.1109/ICRA.2011.5980430
14. B. Hamner, S. Singh, M. Bergerman.: Improving Orchard Efficiency with Autonomous Utility Vehicles. In Proceedings of the 2010 ASABE Annual International Meeting, Pittsburgh, PA, USA, 2010. doi 10.13031/2013.29902
15. J. Moorehead, C. Wellington, J. Gilmore, C. Vallespi.: Automating Orchards: A System of Autonomous Tractors for Orchard Maintenance. In Proceedings of the
16. IEEE Int. Conf. on Intelligent Robots and Systems. Vilamoura, Algarve, 2012.
17. Y. Ye, Z. Wang, D. Jones, et al: Bin-Dog: A Robotic Platform for Bin Management in Orchards. Robotics, 2017. doi 10.3390/robotics6020012
18. Pire, T., Mujica, M., Civera, J., Kofman, E.: The Rosario dataset: Multisensor data for localization and mapping in agricultural environments. The International Journal of Robotics Research, 38(6), 633-641. (2019).
19. Schmuck, P., Chli, M.: CCM-SLAM: Robust and efficient centralized collaborative monocular simultaneous localization and mapping for robotic teams. In: Journal of Field Robotics (JFR). (2018)
20. Umeyama S.: Least-squares estimation of transformation parameters between two point patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 13, pp. 376–380. (1991)

Coordinadores:
- Javier Diaz (UNLP)
- Hugo Ramón (UNNOBA)
- Claudio Aciti (UNCPBA)

CACiC 2023

CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

XII Workshop de Seguridad Informática

> Full Papers

WSI

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Estudio exploratorio sobre la adopción de la Ciberseguridad en PyMES del partido de Hurlingham

Marisa Panizzi[1] [0000-0002-9699-1544], Felipe Ortiz[1] [0000-0003-2291-1924], Iris Sattolo[1] [0000-0002-2132-0033], Fernando Puricelli[1] [0009-0004-5410-4611], Rodolfo Bertone[2] [0000-0003-0609-0310]

[1] Instituto de Tecnología e Ingeniería, Universidad Nacional de Hurlingham, Av. Vergara 2222 (B1688GEZ) - Villa Tesei - Bs. As. Argentina.

[2] Facultad de Informática -Instituto de Investigación en Informática LIDI (III-LIDI). Universidad Nacional de La Plata, 50 y 120. La Plata, Argentina.

marisa.panizzi@unahur.edu.ar;felipe.ortiz@unahur.edu.ar; iris.sattolo@unahur.edu.ar; fernando.puricelli@unahur.edu.ar, pbertone@lidi.info.unlp.edu.ar

Resumen. En Argentina, la industria del software se compone mayoritariamente por Pequeñas y Medianas Empresas (PyMES). Las PyMES requieren incrementar la capacidad y calidad de sus procesos para lograr mayor competitividad dentro de sus sectores industriales. Las soluciones tecnológicas 4.0 enfocadas en la interconectividad, la automatización y los datos en tiempo real contribuyen a tal fin. La adopción de estas tecnologías por parte de las industrias se denomina "industria 4.0[46]. En este sentido, para obtener evidencia sobre el estado actual de la adopción de las tecnologías 4.0 por parte de las PyMES situadas en el partido de Hurlingham se decidió realizar un estudio exploratorio basado en encuesta. Esta primera etapa del estudio se focalizó en la Ciberseguridad en la cual participaron 18 profesionales informáticos que trabajan en PyMES.

Palabras clave: ciberseguridad, PyMES, partido de Hurlingham, encuesta.

1. Introducción

En Argentina, la industria del software se compone mayoritariamente por Pequeñas y Medianas Empresas (PyMES), representando casi el 80% del sector, lo que constituye un eslabón fundamental, en el sector, para el país y refuerza la necesidad de llevar adelante iniciativas que contribuyan con el desarrollo y mejora de competitividad de dichas empresas [1]. Este sector presenta un alto potencial para generar valor agregado al ecosistema productivo del país, promoviendo la generación de empleo calificado y evidenciando un crecimiento exponencial en los últimos años. De hecho, el notable desarrollo queda demostrado en el último informe publicado por el Observatorio de la Economía del Conocimiento (OEC) de Argentina [2].

A nivel internacional se refleja la misma realidad respecto a que las PyMES ocupan una gran porción de la industria de TI [3]. Estas organizaciones reconocieron que es crucial mejorar sus procesos y métodos para ser competitivas en su negocio, aunque no cuentan con recursos y conocimiento suficiente para lograrlo. Se observa que en los

[46] https://www.argentina.gob.ar/produccion/planargentina40/industria-4-0

últimos años las PyMES han surgido muy rápidamente y, en la mayoría de las economías en desarrollo, el sector informático está dominado por pequeñas y empresas emergentes (*Startups*) [4].

De acuerdo con el Ministerio de Desarrollo Productivo en [5] la industria 4.0 se refiere a una nueva manera de producir mediante la adopción de tecnologías 4.0, es decir, de soluciones enfocadas en la interconectividad, la automatización y los datos en tiempo real.

De acuerdo con Rainer Drath y Alexander Horch en [6], las hipótesis o fundamentos que deben ocurrir para que se den las condiciones para el desarrollo de la industria 4.0 son:

- La infraestructura de comunicación en los sistemas de producción será más asequible y por tanto será parte de todo.
- Los dispositivos en el campo, máquinas, plantas y fábricas (incluso productos individuales) estarán más conectados a una red (la Internet o una red privada del fabricante).
- Los dispositivos en el campo, máquinas, plantas y fábricas (incluso productos individuales) serán capaces de almacenar documentos y conocimiento acerca de sí mismos fuera de su corporeidad en la red.

Su primera mención formal con esta connotación data del año 2011, en la Feria de Hannover, Alemania, en la presentación del artículo Industria 4.0: Con el internet de las cosas camino de la 4° revolución industrial[7], donde se expuso cómo Alemania podría ser el próximo líder y proveedor del nuevo mercado en 2020 gracias al internet de las cosas en el entorno industrial.

Para determinar la manera en que las PyMES del partido de Hurlingham pueden fortalecer sus procesos mediante la adopción de tecnologías 4.0[8] se requiere recolectar evidencia sobre el estado actual de la aplicación de la tecnología en sus procesos productivos, evidencias que serán útiles para la que la Universidad pueda reorientar sus recursos de vinculación con el medio socio-productivo de la región. Para lograr esto, nuestro primer estadio de la investigación consiste en realizar un estudio exploratorio a través de una encuesta realizada según las directrices de Molléri *et al*.[9].

El artículo se estructura de la siguiente manera: en la Sección 2 se describe la planificación de la encuesta, en la Sección 3 se describe su ejecución. Los resultados se presentan en la Sección 4. En la Sección 5 se presenta un análisis de las amenazas a la validez y, finalmente, en la Sección 6 se exponen las conclusiones y trabajos futuros.

2. Planificación de la encuesta

Dentro de la planificación de la encuesta, definimos el objetivo principal utilizando la plantilla GQM (*Goal-Question-Metric*) propuesta por Basili *et al*. [10] que se presenta en la Tabla 1. Una vez definido el objetivo global del estudio, en este estadio de la investigación se considera uno de los pilares de la industria 4.0, la adopción de la ciberseguridad por parte de las PyMES del partido de Hurlingham.

Objetivos y preguntas de investigación. En la Tabla 1, se presenta el objetivo de la encuesta mediante la plantilla GQM (*Goal-Question-Metric*) [10].

Tabla 1. Objetivo de la encuesta.

Analizar	La adopción de tecnologías 4.0,
con el propósito de	conocer el estado actual,
con respecto a	descubrir las características de las áreas de conocimiento: el desarrollo de sistemas, la ciberseguridad, la ciencia de datos y la inteligencia artificial; y además el grado de consideración de la ley de género relacionada a la equidad de la distribución de los puestos de trabajo incluyendo los cargos de conducción por parte de estas empresas.
desde el punto de vista	de profesionales informáticos,
en el contexto de	PyMES del partido de Hurlingham.

Las preguntas de investigación (PI) que guiaron esta encuesta son las siguientes:

PI1: ¿Qué características tienen las PyMES del partido de Hurlingham?

P2: ¿Qué características tiene el área de desarrollo de sistemas?

PI3: ¿Cuáles son las características que tiene el área de ciberseguridad?

PI4: ¿Cuáles son las características que tiene el área de inteligencia artificial?

PI5: ¿Qué características tiene el área de ciencia de datos?

PI6: ¿Sería deseable adoptar las tecnologías 4.0?

Proceso de ejecución. Este proceso consta de las siguientes actividades: 1) Diseño de un formulario para llevar un registro sistemático de la ejecución de la encuesta. Este se compone de dos partes: a) Proceso de envío de la encuesta: cantidad de envíos y a quienes. y b) Proceso de seguimiento del envío: mails erróneos, cantidad de cuestionarios respondidos y fecha de respuesta, 2) Envío de la encuesta por correo electrónico. Se diseño un texto de presentación en el cual se menciona el propósito de la investigación, quiénes participan, tiempo estimado para responder la encuesta, agradecimiento por la colaboración y una invitación a que el encuestado difunda la encuesta entre sus contactos, 3) Revisión diaria de encuestas respondidas, 4) Extracción de las respuestas, 5) Revisión de si hay preguntas sin responder y 6) Extracción de los datos.

Población. La población a la cual se decidió enviar la encuesta son PyMES de cualquier sector industrial del partido de Hurlingham. En la Tabla 2 se presenta la clasificación de PyMES utilizada.

Se utilizaron diferentes estrategias para seleccionar la muestra, todas a través de correo electrónico, una de ellas consistió en el envío de la encuesta de manera directa por parte de los autores. La otra estrategia consistió en el envío de manera indirecta a través del Centro PyME-UNAHUR [12].

Con el propósito de lograr una tasa de respuestas más alta, se estableció como estrategia, el envío de recordatorios a que los encuestados a los que se accedió de manera directa.

Diseño de la encuesta. Se diseñó un cuestionario autoadministrado que se envió por correo electrónico a los encuestados y se utilizó la herramienta *Google Forms*. Para el diseño del cuestionario se definieron cinco categorías de preguntas que junto con sus

objetivos se presentan en la Tabla 3. El cuestionario diseñado para las categorías Empresa y área de Ciberseguridad se compone de 24 preguntas de respuestas cerradas y 11 preguntas de respuestas abiertas. Dentro de las preguntas de respuestas cerradas, se emplean respuestas de tipo Sí/No y valores de selección. También se contempla la categoría "Otros" para el caso de que no se cubran explícitamente todas las respuestas posibles. El cuestionario y la correspondencia entre las preguntas del cuestionario con las PI1 y PI3 se encuentran en un apéndice [13] por restricciones de espacio. El cuestionario completo se encuentra disponible en https://forms.gle/qJbUMsSWM27E77eT8

Tabla 2. Clasificación de PyMES en Argentina según la cantidad de empleados [11].

Categoría	Actividad				
	Construcción	Servicios	Comercio	Industria y minería	Agropecuario
Micro	12	7	7	15	5
Pequeña	45	30	35	60	10
Mediana tramo 1	200	165	125	235	50
Mediana tramo 2	590	535	345	655	215

Tabla 3. Categorías del cuestionario y sus objetivos.

Categorías	Objetivos
Empresa.	Determinar las características de las empresas que participan en la encuesta, su sector industrial, la cantidad de empleados y su ubicación geográfica e identificar el perfil del participante de la encuesta, su rol y su antigüedad en el rol. Además, conocer el grado de consideración de la ley de género relacionada a la equidad de la distribución de los puestos de trabajo incluyendo los cargos de conducción vinculados a la tecnología.
Área de desarrollo de sistemas.	Conocer las características de la práctica actual del desarrollo de software en las PyMES de Hurlingham.
Área de ciberseguridad.	Conocer las características de la práctica actual de ciberseguridad en las PyMES de Hurlingham.
Área de inteligencia artificial.	Conocer las características de la práctica actual de la inteligencia artificial en las PyMES de Hurlingham.
Área de ciencia de datos.	Conocer las características de la práctica actual de ciencia de datos en las PyMES de Hurlingham.

Validación. Antes de comenzar con el envío de la encuesta, se realizó una prueba piloto con un grupo reducido de 3 participantes que permitió verificar el tiempo de respuesta inicial estimado de 15 minutos, quedando reducido a 10 minutos. También se logró validar la terminología empleada, los participantes de la prueba piloto confirmaron su claridad y legibilidad. Además, permitió modificar algunas palabras del cuestionario por errores involuntarios de tipeo.

3. Ejecución de la encuesta

Reclutamiento de los participantes. La encuesta se envió de manera indirecta a las PyMES través del Centro PyME-UNAHUR [12] y de manera directa a los contactos de los investigadores.

Gestión de las respuestas. La gestión de la ejecución de la encuesta se realizó de acuerdo con el procedimiento definido en la sección 2. Se realizó un seguimiento diario de las respuestas con el objeto de comprobar que las respuestas estén completas y que las respuestas correspondan a una empresa PyME, en caso de no ser una PyME, se las eliminaba antes del análisis de los datos.

Análisis de los datos. Para asegurar la calidad de los datos obtenidos de la encuesta se revisaron las respuestas con el propósito de encontrar errores de completitud y de tipeo. Se realizó un análisis de contenido de las respuestas de texto libre [14]; se codificaron las respuestas en categorías. También se aplicó el método de teoría fundamentada [15]. Por último, se utilizó un análisis de *clusters* [16] para encontrar relaciones entre los resultados de respuestas de diferentes categorías de la encuesta.

4. Análisis e interpretación de los resultados obtenidos

De las respuestas de los participantes, a continuación, se presentan en primer lugar las características demográficas de los participantes y de las PyMES y, en segundo lugar, los resultados que permitieron dar respuesta a las PI1 y PI3.

La Tabla 4 presenta la distribución de las respuestas por los diferentes roles de los participantes (PE1 y PE1.a.). La mayoría de los participantes son Lideres de proyectos (44,40 %), en segundo lugar, desarrolladores (38,90 %) y, en tercer lugar, gerentes de planta (11,10 %) y un director (5,55%).

Tabla 4. Roles de los participantes.

Rol	Porcentaje
Líder de Proyecto o Jefe de equipo	45,00%
Desarrollador	38,90%
Gerente de planta	11,40%
Director	5,60%

En cuanto a la experiencia (PE2), la mayoría de los participantes cuentan con más de 6 años de experiencia en la industria informática (ver Figura 1).

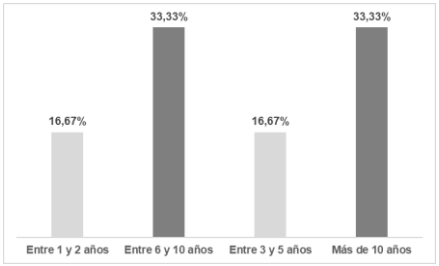


Fig. 1. Experiencia de los participantes en la industria informática.

PI1: ¿Qué características tienen las PyMES del partido de Hurlingham?

Actividad de las PyMES y categorías. La Figura 2 presenta la distribución de respuestas según su tipo de actividad (PE3). Según se muestra en la Figura 3 del total de las PyMES, el 33,33% son micro PyMES, el 22,22% son pequeñas, 27,78% son medianas de tramo 1 y el 16,67% son medianas de tramo 2 (PE5).

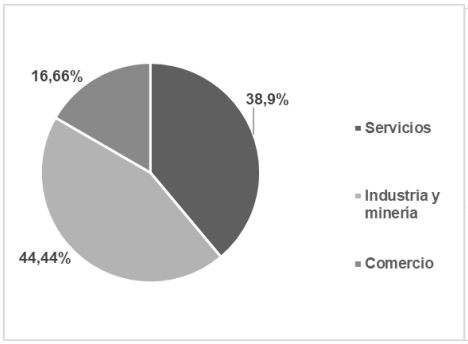


Fig. 2. Distribución de PyMES por actividad.

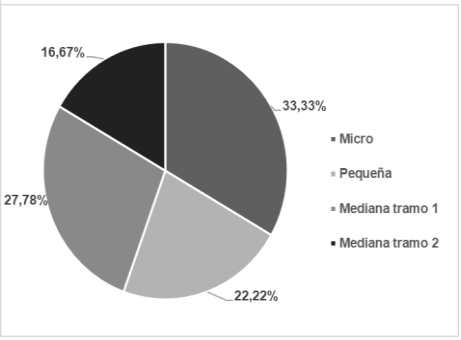


Fig. 3. Distribución de PyMES por tamaño.

Cantidad de empleados, cantidad de empleadas mujeres, mujeres vinculadas a la tecnología y roles. La mayoría de los empleados son de género masculino, es muy baja la cantidad de empleados de género femenino (PE4 y PE5). Solo el 16,67% de las PyMES cuentan con personal de género femenino cuyos roles están vinculados a la tecnología y el rol desempeñado es el de desarrollador (PE6, PE6.a y PE6.b).

PI3: ¿Cuáles son las características que tiene el área de ciberseguridad?

Estrategia de seguridad. La Figura 4 presenta que el 16,67% de las PyMES cuentan con una estrategia de seguridad documentada (PE7). Estas PyMES son de tramo 2 y además comunican la estrategia a los participantes clave de la empresa (PE7.a).

Métricas y riesgos. En relación a las métricas, ninguna de las PyMES cuenta con ellas (PE8), tampoco cuentan con un plan integral de concientización (PE9) y no realizan la identificación de los riesgos a los que la PyME se encuentra expuesta (EP10). Solamente el 16,67% de las PyMES de tramo 2 realizan la evaluación de impacto empresarial como parte de la evaluación de riesgos (PE11) utilizando una matriz FODA (PE11.a.).

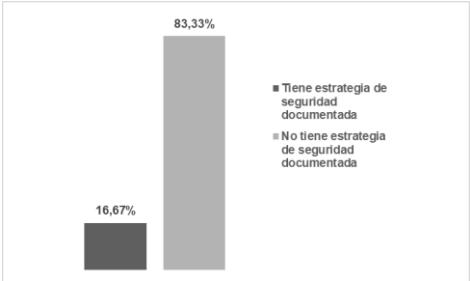


Fig. 4. PyME cuenta o no con una estrategia de seguridad documentada.

Entornos y servidores. La Figura 5 presenta los entornos que se encuentran segregados (PE12 y PE12.a.) resultando el más seleccionado por los encuestados, es el de pruebas. En relación con si los servidores se encuentran protegidos contra ataques maliciosos, la mayoría de los encuestados respondieron afirmativamente (PE12) (Ver Figura 6).

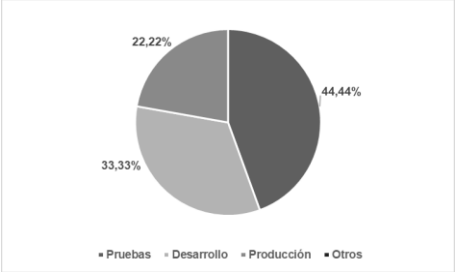


Fig. 5. Entornos segregados.

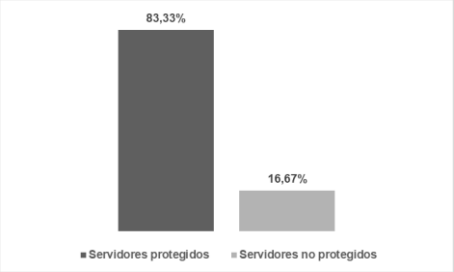


Fig. 6. Servidores protegidos.

Acceso y resguardo de la información. Los encuestados del total de las PyMES coincidieron en que los usuarios siempre se autentican utilizando un identificador (por ejemplo, un ID de usuario) y un autenticador (por ejemplo, una contraseña) (PE15).

En relación con si la PyME realiza copias de seguridad de la información esencial y del software, todos los encuestados respondieron afirmativamente (PE16). Del total de las PyMES, el 83,33% pueden en caso de emergencia restaurar la información esencial o el software dentro de los plazos críticos (PE17). La mitad de las PyMES tienen sistemas que cuentan con software de protección contra *malware* (PE18). Del total de las PyMES que participaron del estudio solo el 16,67% cuentan con los sistemas y las redes monitoreadas para identificar posibles brechas de seguridad (PE19) y el 27,78% cuentan con un plan de continuidad de negocios (EP20) pero no realizan pruebas de estos planes (PE20.a.). Respecto a si las PyMES cuentan con medidas de protección ante robos, solo el 33,33% cuentan con estas (PE21), el 16,67%3 de las PyMES cuentan con medidas de protección de la documentación y el otro 16,67% de las PyMES cuentan con medidas de protección de la documentación y de los equipos (PE21.a.). La mitad de las PyMES cuentan con información sensible almacenada en medios físicos protegidos contra la pérdida o divulgación no autorizada (PE22).

Intereses de los encuestados. Del total de los encuestados, el 66,67% de las PyMES manifestaron interés por recibir información relacionada a capacitaciones o asesoramiento en aspectos de seguridad para su PyME por parte de la Universidad. Esto permitió al Centro PyME-UNAHUR reorientar las actividades previstas en el plan estratégico del área para satisfacer las necesidades de las PyMES.

5. Amenazas a la validez

Para analizar la validez de la encuesta se tuvieron en cuenta los tipos de validez propuestos por Wohlin *et al.* [17]:

- Validez de conclusiones. El tamaño de la muestra (18 encuestas) se considera pequeño para considerar los resultados estadísticamente aceptables. Si bien se

reconoce que es aconsejable ampliar la muestra, pero por tratarse de un estudio exploratorio sobre la adopción de las tecnologías 4.0 específicamente la ciberseguridad permitió cumplir con el propósito definido para esta encuesta. Además, se aplicó el proceso de manera sistemática y rigurosa para permitir que el proceso sea reproducible.

- Validez interna. Los principales problemas que afectan la validez interna de nuestro estudio se refieren al encuadre y al muestreo de los participantes. Nuestra estrategia de reclutamiento de participantes podría haber sido sesgada por una alta probabilidad de similitud de perfiles y por profesionales sin experiencia. Si bien hay una variedad en los roles, la mayoría de los encuestados cuentan con más 6 años de experiencia en la industria informática. Otro factor negativo podría haber sido la dificultad para comprender las preguntas (por ejemplo, ambiguas, poco claras, mal formuladas), esto quedó resuelto con la prueba piloto realizada con 3 encuestados. Las motivaciones de los encuestados también podrían haber afectado las respuestas y, por lo tanto, los resultados obtenidos del estudio. Esto quedó resuelto porque en el mail que se envió junto con la encuesta quedó explicito el compromiso de compartir los resultados del estudio con los participantes. Para evitar que hubiesen más de una respuesta por empresa, se revisaron cuidadosamente los contactos y las respuestas obtenidas de los envíos indirectos.
- Validez externa. Se seleccionaron a los participantes de la encuesta de forma de que sean profesionales de la industria informática que se desempeñen en PyMES de Argentina de la región de Hurlingham. Esto permitió realizar observaciones sobre el estado actual de adopción de tecnologías 4.0.
- Validez de constructo. Se definieron las preguntas de investigación de manera cuidadosa, así como también el esquema de las categorías de preguntas y los posibles valores de las respuestas. Todo el proceso ha sido consensuado entre los investigadores de manera que no sea sesgado el objeto de estudio. Otros aspectos que permitieron disminuir esta amenaza han sido que, en el mail de la invitación a los participantes, se les explicó claramente el propósito del estudio y del propio cuestionario se visualiza su anonimato salvo que el participante estuviese interesado en que lo podamos contactar para profundizar el cuestionario.

6. Conclusiones y trabajos futuros

Este artículo presenta los resultados de una encuesta realizada sobre el estado actual de la adopción de tecnologías 4.0 por parte de las PyMES del partido de Hurlingham, específicamente en ciberseguridad a la que respondieron 18 profesionales de la industria informática que trabajan en PyMES de Argentina de la región de Hurlingham. Los principales hallazgos obtenidos para las PI1 y PI2 se detallan a continuación:

PI1: ¿Qué características tienen las PyMES del partido de Hurlingham?

Del total de las PyMES, la mayoría de los encuestados que participaron en el estudio se distribuyen entre los roles de Líder de proyecto o jefe de equipo (44,40%) y desarrolladores (38,90%) y con una experiencia mayor a 6 años en la industria informática. Del total de las PyMES, el 33,33% son micro PyMES, el 22,22% son

pequeñas, el 27,78% son medianas de tramo 1 y el 16,67% son medianas de tramo 2. Se logró evidenciar que no hay equidad en la distribución de los puestos de trabajo, la mayoría de los empleados son del género masculino y que el único rol desempeñado por empleados del género femenino vinculado a la tecnología es el de desarrollador, esto indica que no hay empleadas mujeres que desempeñen roles gerenciales referidos a la tecnología.

PI3: ¿Cuáles son las características que tiene el área de ciberseguridad?

La información es un recurso que, como el resto de los activos de una organización, tiene valor para una PyME y por consiguiente debe ser debidamente protegida frente a amenazas que podrían afectar la confidencialidad, integridad, disponibilidad, auditabilidad y legalidad de esta.

De las PyMES que participaron en la encuesta, solo el 16,67% cuentan con una estrategia de seguridad documentada exponiendo a la seguridad de la información (necesaria para la continuidad de sus operaciones). Algunos de los resultados más alarmantes son los siguientes:

- Ninguna de las PyMES identifica los riesgos a los que se encuentra expuesta.
- Ninguna de las PyMES cuenta con un plan integral de concientización
- En general, no existe una adecuada segregación de entornos y servidores.
- Solo la mitad de las PyMES tienen sistemas que cuentan con software de protección contra *malware*.
- Del total de las PyMES que participaron en la encuesta, solo 3 cuentan con los sistemas y las redes monitoreadas para identificar posibles brechas de seguridad.
- La mitad de las PyMES cuentan con información sensible almacenada en medios físicos protegidos contra la pérdida o divulgación no autorizada.

A partir de los resultados de esta encuesta, se confirma la necesidad de: a) ampliar la muestra, b) continuar con la obtención de resultados para dar respuesta a las PI2, PI4, PI5 y PI6 que forman parte de la encuesta global sobre los pilares de la industria 4.0 y por último, c) contrastar los resultados obtenidos del partido de Hurlingham con otros partidos.

Referencias

1. Reporte anual 2018 sobre el Sector de Software y Servicios Informáticos de la República Argentina. OPSSI. Disponible en: https://www.cessi.org.ar/opssi (2018).
2. Ministerio de Producción y Trabajo. Presidencia de la Nación. Secretaría de la Transformación Productiva. Informe Argentina Productiva - Economía del Conocimiento.
3. Disponible en: https://biblioteca.produccion.gob.ar/buscar/?fid=16 (2019).
4. Hisham M. Abushama. PAM-SMEs: process assessment method for small to mediumenterprises. Software: Evolution and Process, 28, pp. 689 –711 (2016).
5. Sharma P., Sangal A.L. Building a hierachical structure model of enablers that affect software process improvement in software SMEs-A mixed method approach. Computer Standards & Interfaces, 66, pp. 1–23 (2019).
6. "¿Qué es la Industria 4.0?", Argentina.gob.ar, el 7 de abril de 2021. https://www.argentina.gob.ar/produccion/planargentina40/industria-4-0 (consultado el 28 de junio de 2023).

7. R. Drath y A. Horch, “Industrie 4.0: Hit or Hype? [Industry Forum]”, IEEE Ind. Electron. Mag., vol. 8, núm. 2, pp. 56–58, jun. 2014, doi: 10.1109/MIE.2014.2312079.
8. “Industrie 4.0: Mit dem Internet der Dinge auf dem Weg zur 4. industriellen Revolution - ingenieur.de”, ingenieur.de - Jobbörse und Nachrichtenportal für Ingenieure, el 1 de abril de 2011. https://www.ingenieur.de/technik/fachbereiche/produktion/industrie-40-mit-internet-dinge-weg-4-industriellen-revolution/ (consultado el 20 de junio de 2023).
9. Ministerio de Desarrollo Productivo (2021). Plan de Desarrollo Productivo Argentina 4.0. Disponible en: https://www.argentina.gob.ar/produccion/planargentina40/industria-4-0.
10. Molléri J.S., K. Petersen K., E. Mendes E. An empirically evaluated checklist for surveys in software engineering. Information and Software Technology 119 (2020) 106240 (2020).
11. Basili V., y Rombach D. The TAME Project: Towards Improvement-Oriented Software Environments. IEEE Transactions on Software Engineering, 14(6) (1988).
12. Ministerio de Desarrollo Productivo. Nuevas categorías para ser PyMES. Disponible en: https://www.argentina.gob.ar/noticias/nuevas-categorias-para-ser-pyme (2018).
13. Centro PyME-Unahur. Vinculación Tecnológica de la Universidad Nacional de Hurlingham. Disponible: https://unahur.edu.ar/centro-pyme-unahur/
14. Panizzi M. Apéndice. Estudio exploratorio sobre la adopción de la Ciberseguridad en PyMES del partido de Hurlingham. Disponible en: https://doi.org/10.6084/m9.figshare.23657181.v1
15. Krippendorff K. Content analysis: An introduction to its methodology, 3rd edn. Sage Publications, Thousand Oaks (2012).
16. Corbin J, Srtauss, A. Basics of qualitative research; techniques and procedures for developing grounded theory. Sage (2008).
17. Matloff N. From Algorithm to Z-Scores: Probabilistic and Statistical Modeling in Computer Science. Disponible en: http://heather.cs.ucdavis.edu/~matloff/probstatbook.html (2009).
18. Wohlin C., Runeson P., Höst M., Ohlsson MC., Regnell B., Wesslén A. Experimentation in software engineering: an introduction. Springer (2012).

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Cifrador de Bloque con Alternancia de Matrices de Permutación y Combinación de Algoritmos Cifradores Paralelos

Andrés Francisco Farías – Andrés Alejandro Farías

Departamento Académico de Ciencias Físicas, Matemáticas y Naturales
Universidad Nacional de La Rioja, La Rioja. Argentina
afarias665@yahoo.com.ar, andres_af86@hotmail.com

Abstract. Cifrador de bloque, desarrollado sobre la estructura de una red de Feistel, de tamaño de bloque de 256 bits, con clave de 128 bits y vector de inicialización de 256 bits, con alternancia de matrices de permutación y combinación de algoritmos cifradores paralelos, con modo de encadenamiento de bloques de cifrado de propagación (PCBC, Propagating Cipher Block Chaining).

El primer algoritmo contiene un generador binario pseudoaleatorio que está compuesto por cuatro Linear Feedback Shift Registers (LFSR), de distintas longitudes, cuyas secuencias se combinan mediante una operación de suma con acarreo. El segundo algoritmo contiene un generador binario pseudoaleatorio que está conformado por tres LFSR, de distintas longitudes, cuyas secuencias se combinan mediante una suma con acarreo.

Los algoritmos trabajan en paralelo y sus secuencias se combinan mediante una operación XOR.

Los LFSR de cada uno de los generadores contienen funciones booleanas de filtrado no lineal de cuatro variables. Las funciones fueron seleccionadas por sus propiedades criptográficas tales como: ser balanceadas, alto grado algebraico, cumplir con el Criterio de Avalancha Estricta (SAC, sigla en inglés) y tener alta no linealidad. Finalmente, el texto cifrado obtenido fue sometido a un conjunto de pruebas estadísticas de aleatoriedad.

Keywords: LFSR, cipher, key, boolean function, non-linearity

1. Introducción

El presente documento expone el desarrollo de un cifrador de bloque, basado en una Red de Feistel que permite el cifrado y descifrado utilizando la misma estructura, donde para el caso del descifrado se utilizan las subclaves cambiando el orden de las mismas [1], [2]. La clave adoptada es de 16 caracteres, es decir 128 bits.

2. Esquema del cifrador

El cifrado de bloque se denomina así por realizar el proceso de encriptación trabajando sobre cadenas de texto de igual longitud. En este caso se utilizaron bloques de 256 bits. Luego esos bloques son ensamblados siguiendo el modo de encadenamiento de bloques de cifrado de propagación (Propagating Cipher Block Chaining. (PCBC)). Básicamente la estructura del cifrador está conformada por; Red

de Feistel que consta de: Red de Feistel para cifrado, Red de Feistel para descifrado, Clave y subclaves, Vector de inicialización, Algoritmos de cifrado, Generadores binarios pseudoaleatorios, Modo de operación PCBC, Matrices de Permutación

3. Red de Feistel para cifrado

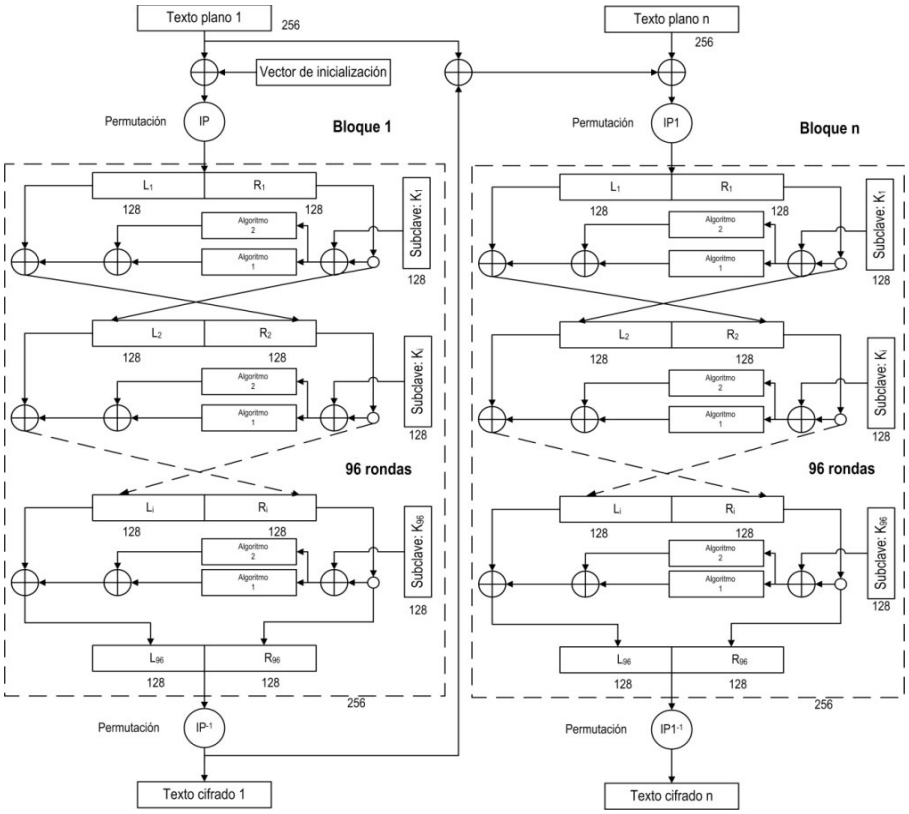


Fig. 1. Red de Feistel para cifrado

El proceso de cifrado consiste en dividir el texto plano en bloques de 256 bits, el primer bloque es sometido a una operación XOR con el vector de inicialización, luego al resultado se le realiza una permutación IP.

La salida de la permutación entra en la red de Feistel, que se detalla en la figura 1 y se producen 96 rondas, con sus respectivas subclaves, después se realiza una permutación IPI, para obtener el primer bloque de texto cifrado.

Para los siguientes bloques de texto plano, se realiza una operación XOR con los bloques de texto plano y cifrado del primer bloque y al resultado se le ejecuta una nueva operación XOR con el texto plano del bloque y la salida sufre una permutación IP1 (la otra matriz de permutación) antes de entrar a la red de Feistel y producir 96 rondas, con las subclaves correspondientes..

Después de esta operación se calcula la permutación IPI1 (la otra matriz de permutación inversa) y se consigue un nuevo bloque de texto cifrado. Las matrices de permutación se van alternando de un bloque a otro.

4. Red de Feistel para descifrado

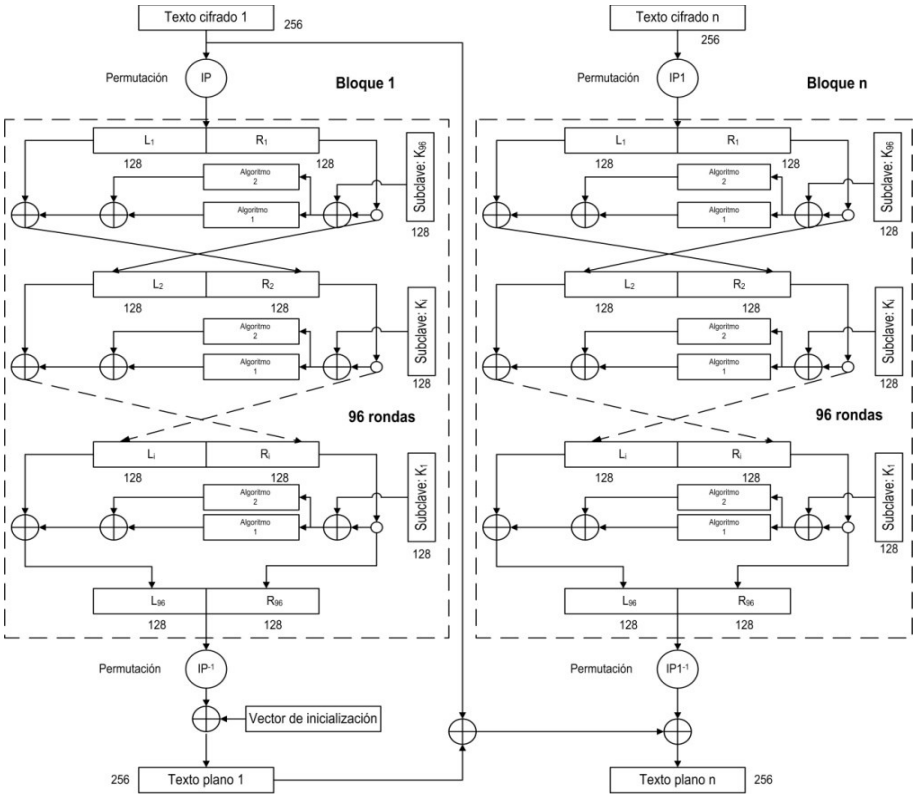


Fig. 2. Red de Feistel para descifrado

La Red de Feistel para descrifrado es similar a la anterior, pero en este caso se toma el texto cifrado y se lo divide en bloques de 256 bits.

Para el primer bloque de texto cifrado se realiza una permutación IP antes de entrar a la red de Feistel y realizar 96 rondas, con las claves introducidas en modo inverso, al resultado se le realiza una permutación IPI y luego se produce una operación XOR con el vector de inicialización para obtener el primer bloque de texto plano.

Para el resto de los bloques de texto cifrado, el proceso comienza con la permutación IP1, después se ingresa a la red de Feistel y se llevan a cabo 96 rondas, con las subclaves ingresadas en modo inverso.

Finalmente después de este proceso se hace una permutación IPI1 y a la salida se le aplica una operación XOR con la resultante de la operación XOR entre el texto cifrado

y texto plano del bloque anterior, para lograr un nuevo bloque de texto plano. Las matrices de permutación se van alternando de un bloque a otro.

5. Clave y subclaves

Como se dijo previamente, la clave está conformada con 16 caracteres (128 bits), de la que se generan 96 subclaves de 128 bits, siguiendo los pasos que se muestran en la figura 3.

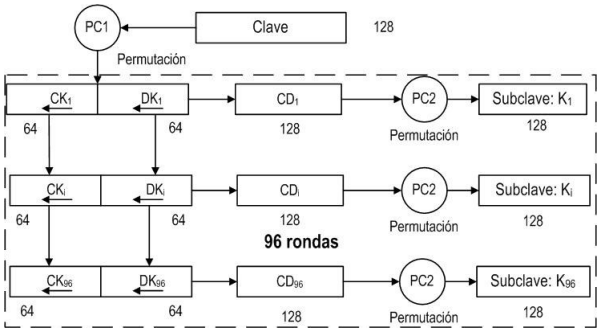


Fig. 3. Tratamiento de las subclaves

La clave es sometida a una permutación según la matriz de permutación PC1; luego se divide el bloque de 128 bits resultante en dos bloques de 64 bits, los que sufren desplazamiento de las posiciones de los bits de manera de tener 96 pares de bloques de 64 bits que corresponderán a las 96 subclaves.

Esos pares son ensamblados y luego sometidos a la permutación PC2, para obtener las 96 subclaves finales.

6. Vector de inicialización

Es para iniciar las tareas de encadenamiento de bloques, tanto de cifrado como de descifrado. Es única para todo el proceso, debe ser secreta como la clave y su longitud es igual a la de los bloques: 256 bits.

7. Algoritmos de cifrado

Los algoritmos de cifrado tienen la configuración que se indica en la figura 4, pero difieren en el generador binario pseudoaleatorio.que es distinto para cada uno.

Tienen una entrada de 128 bits, que conforman los estados iniciales para los LFSR del generador, que una vez cargados, realizan 128 ciclos, entregando 128 bits de salida.

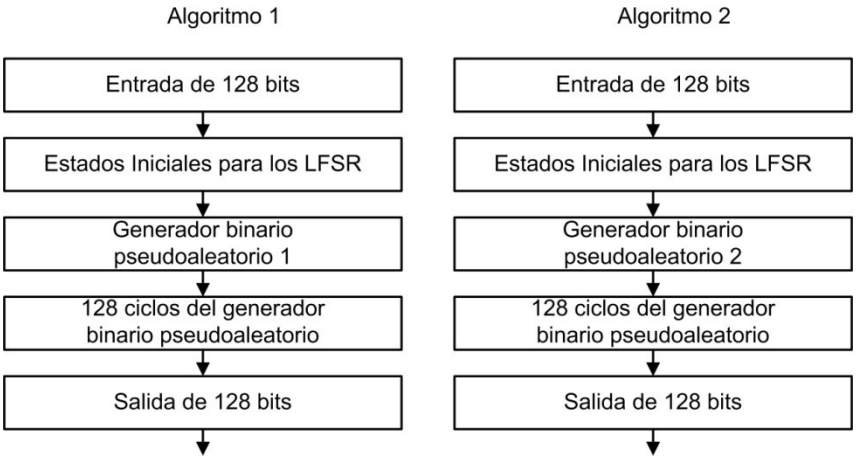


Fig. 4. Algoritmos de cifrado

8. Generadores binarios pseudoaleatorios

Está conformado por cuatro y tres LFSR con funciones booleanas de filtrado no lineal de cuatro variables, balanceadas y de alta no linealidad. Luego se combinan, mediante suma con acarreo, para obtener la secuencia final.

8.1 Características de los LFSR

Los LFSR que se utilizan tienen la siguiente estructura, un LFSR de base con un polinomio de conexión primitivo que produce el cálculo de la retroalimentación. En paralelo tenemos una función booleana de filtrado no lineal de cuatro variables, que se alimenta con cuatro valores de los registros del LFSR. La salida de la función y la retroalimentación del polinomio se someten a una operación XOR para obtener la retroalimentación definitiva del LFSR.

8.2 Definición de los generadores

Para los dos generadores se adoptó la misma estructura, cambiando entre ellos las longitudes de los LFSR y las funciones booleanas de filtrado no lineal.

Básicamente se utilizan para el primer generador cuatro LFSR con funciones booleanas de filtrado no lineal, cuyas secuencias se combinan mediante suma con acarreo, figura 5

Para el segundo generador se dispone de tres LFSR, también con funciones booleanas de filtrado no lineal, de los que se obtiene un secuencia mediante la aplicación de la suma con acarreo, figura 5

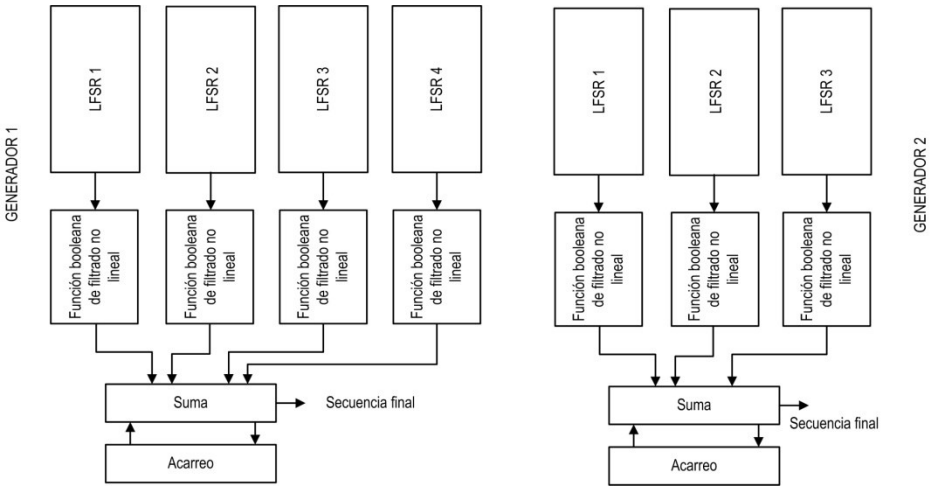


Fig. 5. Esquema generador binario pseudoaleatorio 1 y 2

8.3 Elección de los distintos LFSR

En tabla se indican las longitudes y polinomios primitivos de cada uno de ellos [3], [4], [5].

Tabla 1. LFSR, longitudes y polinomios primitivos del generador

LFSR	Longitud	Polinomios primitivos
1	71	$P(x)_1 = x^{71} + x^{59} + x^{53} + x^{48} + 1$
2	67	$P(x)_2 = x^{67} + x^{61} + x^{33} + x^3 + 1$
2	61	$P(x)_3 = x^{61} + x^{44} + x^{19} + x^{15} + 1$
3	59	$P(x)_4 = x^{59} + x^{54} + x^{46} + x^{26} + 1$

8.4 Propiedades criptográficas deseables

A continuación se indican algunas de las propiedades criptográficamente más significativas, adoptadas para este trabajo [6], [7], [8].

- Función Balanceada:
- No Linealidad:
- Grado Algebraico:
- SAC: Tabla de funciones para los generaadores

Siguiendo los criterios arriba indicados las funciones booleanas aceptadas, para los generadores, por tener buenas propiedades criptográficas, se enumeran a continuación:

Tabla 2. Funciones aceptadas para el generador 1

f_{NAF}	
$f_{5775} = a \oplus b \oplus a \cdot b \oplus a \cdot c \oplus a \cdot d$	$f_{1550} = b \oplus a \cdot b \oplus a \cdot c \oplus b \cdot d \oplus c \cdot d$
$f_{4529} = a \oplus c \oplus a \cdot c \oplus b \cdot c \oplus c \cdot d$	$f_{1579} = b \oplus a \cdot b \oplus a \cdot c \oplus a \cdot d \oplus b \cdot c \cdot d$

$f_{2402} = b \oplus a \cdot c \oplus b \cdot c \oplus d \oplus c \cdot d$ $\quad$ $f_{1585} = b \oplus a \cdot b \oplus a \cdot c \oplus d \oplus a \cdot d$

$f_{3338} = a \oplus a \cdot b \oplus c \oplus b \cdot c \oplus b \cdot d$

En figura 6, se indica la conformación final de las generadores 1 y 2.

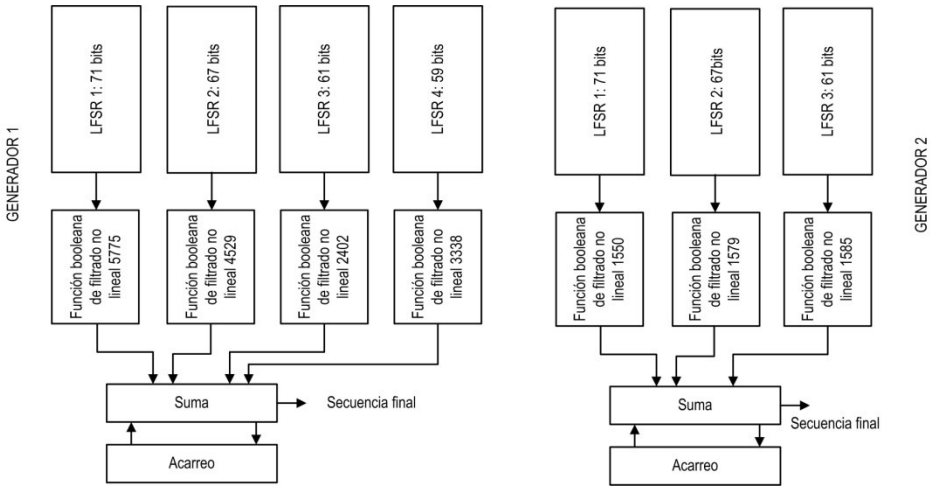


Fig. 6. Esquema generador pseudoaleatorio 1 y 2

9. Modo de operación:

Modo de encadenamiento de bloques de cifrado de propagación (Propagating Cipher Block Chaining.(PCBC))

10. Matrices de Permutación

Se recurre a matrices distribución aleatoria de las posiciones, para obtenerlas se utiliza un generador de números aleatorios, en esta ocasión se adopta un generador congruencial multiplicativo[7]. El generador tiene la siguiente expression

$$x_{i+1} = (a_x \cdot x_i)mod\ m_x$$

Donde: $a_x = multiplicador\ \ m_x = módulo\ \ x_0 = semilla$

Tabla 3. Tabla.6. Matriz IP

Matriz	módulo	multiplicador	semilla
IP	1048576	2011	2047
IP1	1048576	3421	3571
PC1	1048576	1297	1229
PC2	1048576	1789	1663

11. Pruebas de aleatoriedad

11.1 Elección de las pruebas estadísticas

Fueron seleccionadas algunas pruebas de la Norma NIST Special Publication 800-22, del trabajo de Rukhin (et al.) [9]: Prueba de frecuencia, Prueba de frecuencia en un bloque, Prueba de rachas, Prueba de rachas de unos en un bloque, Prueba de sumas acumuladas, Prueba de entropía aproximada

11.2 Pruebas sobre el generador

Se analizaron cien secuencias binarias de 100.000 bits, obtenidas del generador a partir de cien claves distintas. El nivel de significancia adoptado para las pruebas estadísticas es de $\alpha = 0{,}01$. La hipótesis nula es: $H_0 \rightarrow p\_valor > 0{,}01$

11.3 Proporción de muestras que pasan las pruebas

Para el análisis de los resultados, se determina la proporción de muestras que superan las pruebas, y con esos datos se construye un gráfico de puntos, luego se verifica si los mismos caen dentro de los límites superior e inferior, donde k es el número de muestras. $LS, LI = (1-\alpha) \pm 3 \cdot \sqrt{\alpha \cdot (1-\alpha)/k}$

En nuestro caso $k = 100$ y el nivel de significancia elegido es: $\alpha = 0.01$, los límites quedan: $LS = 1{,}02$ y $LI = 0{,}96$

Se consideran todas pruebas, los resultados se indican en la tabla.

Tabla 4. Pruebas

Pruebas	Proporción	Superior	Inferior
Frecuencias	1,00	1,02	0,96
Frecuencias en un Bloque	0,99	1,02	0,96
Rachas	1,00	1,02	0,96
Rachas de Unos en un Bloque	0,99	1,02	0,96
Sumas Acumuladas Adelante	1,00	1,02	0,96
Sumas Acumuladas Atrás	1,00	1,02	0,96
Entropía Aproximada	1,00	1,02	0,96

En figura 7, se aprecia el resultado, en definitiva, la secuencia que entrega el generador supera las pruebas de aleatoriedad.

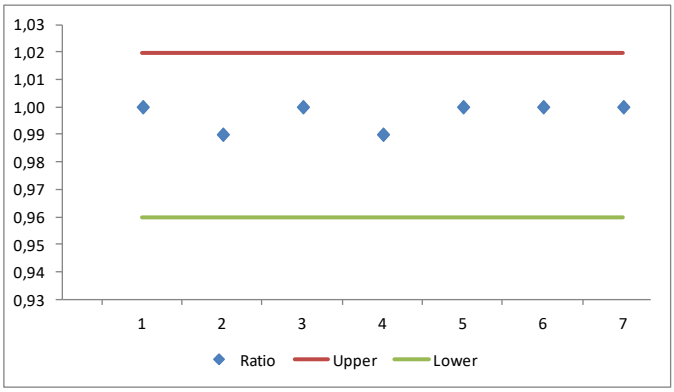


Fig. 7. Gráfico de puntos

12. Comparación de frecuencias

Superposición de gráficos de frecuencias para observar las diferencias entre texto plano y texto cifrado, figura 8.

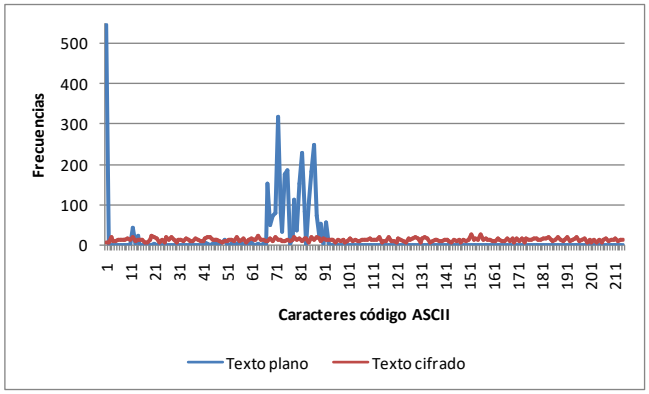


Fig. 8. Frecuencias de caracteres del texto plano y cifrado

13. Conclusiones

Un cifrador de bloque de 256 bits, con algunas propiedades importantes tales como clave de 128 bits y vector de inicialización de 256 bits y la incorporación de algoritmos de cifrado que contienen generadores binarios pseudoaleatorios.

Para futuras versiones se pueden incorporar entre otras cosas: claves más largas y mayor cantidad de algoritmos y otros métodos de concatenación de bloques.

El resultado obtenido del texto cifrado tiene una frecuencia de caracteres aleatorios, lo que hace difícil un criptoanálisis basado en la estadística de aparición de caracteres.

Referencias

1. Karakoç, F., Demirci, H., Harmanc, A.: AKF: A Key Alternating Feistel Scheme for Lightweight Cipher Designs, Information Processing Letters. 115, 359--367 (2015)
2. Bogdanov, A.: Analysis and Design of Block Cipher Constructions. Fakultat fur Elektrotechnik und Informationstechnik an der Ruhr-Universitat Bochum (2009)
3. Clark, J., Jacob, J., Maitra, S., Stanica, P.: Almost Boolean Functions: The Design of Boolean Functions by Spectral Inversion. Computational intelligence. 20. (3), 450—462 (2004)
4. Braeken, A.: Cryptographic Properties of Boolean Functions and S-Boxes. Faculteit Ingenieurswetenschappen. Katholieke Universiteit Leuven (2003)
5. Elhosary, A., Hamdy, N., Farag, I., Rohiem, I.: State of the Art in Boolean Functions Cryptographic Assessment. International Journal of Computer Networks and Communications Security.1. (3), 88--94 (2013)
6. García Méndez, P.: Descripción Polinomial de los Sistemas de Cifrado DES y AES. Universidad Autónoma Mexicana, México (2011)
7. Fishman, G.: Multiplicative Congruential Random Number Generators with Modulus 2ß: An Exhaustive Analysis for ß = 32 and a Partial Analysis for ß = 48. Mathematics of Computation. 54. (189), 33--344 (1990)
8. Duta, C., Mocanu, B., Vladescu, F., Gheorghe, L.: Randomness Evaluation Framework Of Cryptographic Algorithms. International Journal on Cryptography and Information Security (IJCIS), Vol. 4, No. 1, (2014)
9. Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S., Levenson, M., Vangel, M., Banks, D., Heckert, A., Dray, J., and Vo, S., “A Statistical Prueba Suite for Random and Pseudorandom Number Generators for Cryptographic Applications”, National Institute of Standards and Technology, (2000).

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Simulación de Montecarlo, con Dos Variables Aleatorias, Utilizando Generador Binario Pseudoaleatorio, para Problemas de Reparación de Máquinas

Andrés Francisco Farías[1], Germán Antonio Montejano[2], Ana Gabriela Garis[3], Pablo Marcelo García[4], Andrés Alejandro Farías[5]

Universidad Nacional de La Rioja, La Rioja [1,5] – Argentina,
Universidad Nacional de San Luis, San Luis – Argentina [2,3],
Universidad Nacional de La Pampa, Santa Rosa – Argentina [4]

afarias665@yahoo.com.ar [1], gmonte@unsl.edu.ar [2], agaris@gmail.com [3], pablogarcia@unlpam.edu.ar [4], andres_af86@hotmail.com

Abstract. Simulación de Montecarlo de dos variables aleatorias, con números aleatorios entregados por un generador binario. Se trabaja sobre un caso de reparación de máquinas similares de una planta industrial, donde se realizaron 200 observaciones de los tiempos entre fallas y el tiempo que demandó la reparación de las máquinas. Para ello se ejecuta un experimento para un ciclo de 20 fallas y luego se repite el procedimiento 200 veces, de donde se obtiene el promedio de las mismas.

Para realizar esta simulación de Montecarlo se necesitan obtener números aleatorios, para lograrlo podemos contar con distintos dispositivos, uno de ellos es el generador de secuencias binarias pseudoaleatorias.

En este trabajo se expone el procedimiento de diseño de dicho mecanismo para producir secuencias binarias, basadas en la combinación de registros de desplazamiento con retroalimentación lineal (Linear Feedback Shift Register, LFSR). El proceso incluye la descripción del modelo, la estructura de cada generador, selección de las funciones booleanas que cuenten con las mejores propiedades criptográficas, la definición de la combinación final. Luego se verifica la aleatoriedad de las secuencias obtenidas, se aplican a las mismas un conjunto de pruebas estadísticas de aleatoriedad. Luego la secuencia binaria se convierte en secuencia decimal, mediante un protocolo específico.

Keywords: LFSR, cipher, key, boolean function, non-linearity, simulation

1. Introducción

El trabajo consta de siguientes etapas:

- Diseño del generador binario pseudoaleatorio
- Pruebas de aleatoriedad
- Conversión de la secuencia binaria a decimal
- Simulación de Montecarlo de dos variables aleatorias
- Caso práctico: Reparación de máquinas.

2. Generador binario pseudoaleatorio

Entre otras características de este generador tenemos la facilidad de implementación, pero, fundamentalmente un período con una longitud significativa.

Es en esos términos que se propone un modelo que responda a tales exigencias. La modalidad elegida se basa en la combinación no lineal de secuencias producidas por cuatro LFSR [1], [2].

El procedimiento de construcción de un generador pseudoaleatorio de ese estilo requiere de varias etapas:

- Definición esquemática del modelo.
- Elección de los distintos LFSR.
- Selección de funciones booleanas en base a sus propiedades criptográficas.
- Conformación del generador con los componentes ya seleccionados.
- Clave y el procedimiento para generar los estados iniciales de los LFSR.
- Pruebas estadísticas y los criterios de interpretación de los resultados.
- Puesta en funcionamiento y realización de las pruebas de aleatoriedad.

2.1 Definición esquemática del modelo

El generador propuesto en este trabajo, está conformado por cuatro LFSR, que tienen cada uno, una función booleana de filtrado no lineal, que producen secuencias binarias, las que luego son combinadas mediante otras funciones booleanas de combinación, y una operación final XOR.:

2.2 Elección de los distintos LFSR

Las longitudes y polinomios primitivos de cada LFSR, que componen el generador, son las siguientes [3], [4], [5], en la tabla 1 y en la figura 2 se indican como queda el dispositivo.

Tabla 5. LFSR, longitudes y polinomios primitivos del Generador

LFSR	Longitud	Polinomios primitivos
1	47	$P(x) = x^{47} + x^{32} + x^{24} + x^{11} + 1$
2	61	$P(x) = x^{61} + x^{57} + x^{26} + x^{3} + 1$
3	59	$P(x) = x^{59} + x^{54} + x^{46} + x^{26} + 1$
4	53	$P(x) = x^{53} + x^{50} + x^{41} + x^{20} + 1$

2.3 Selección de las funciones booleanas

Propiedades criptográficas deseables. A continuación se indican algunas de las propiedades criptográficamente más significativas, adoptadas para este trabajo [6], [7], [8]: Función Balanceada, Alta No Linealidad, Alto Grado Algebraic y Criterio de Avalancha Estricto.

Siguiendo los criterios arriba indicados las funciones booleanas aceptadas, son observadas en tabla 2:

Tabla 6. Funciones de cuatro variables adoptadas

$f_{5775} = a \oplus b \oplus a \cdot b \oplus a \cdot c \oplus a \cdot d$
$f_{4529} = a \oplus c \oplus a \cdot c \oplus b \cdot c \oplus c \cdot d$
$f_{4722} = a \oplus b \oplus a \cdot c \oplus b \cdot c \oplus c \cdot d$
$f_{4393} = a \oplus c \oplus a \cdot d \oplus b \cdot d \oplus c \cdot d$
$f_{3981} = a \oplus a \cdot c \oplus b \cdot c \oplus d \oplus c \cdot d$
$f_{3672} = a \oplus a \cdot b \oplus c \oplus a \cdot c \oplus a \cdot d$

2.4 Conformación del generador combinacional

El generador combinacional se indica en la figura 1:

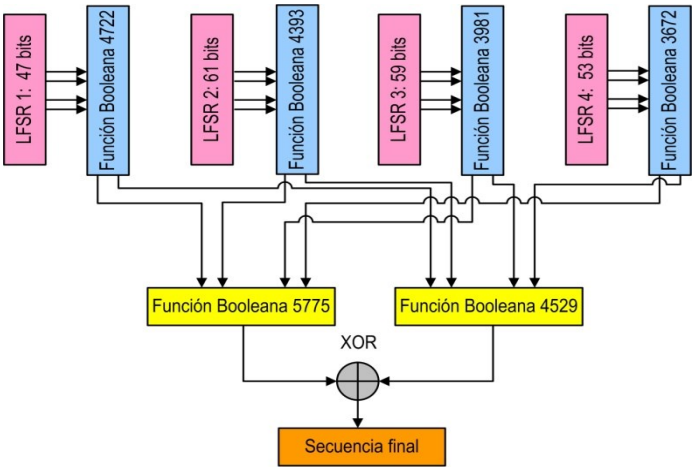


Fig. 1. Generador Combinacional

2.5 Clave

Para originar los estados iniciales de los distintos LFSR se realiza un proceso que utiliza una clave de 32 caracteres, que expresada en código ASCII (American Standard Code for Information Interchange), tiene longitud de 256 bits.

La clave es sometida a un proceso criptográfico, De la operación resulta un vector SK[j] de 256 bits, que es el que proveerá los estados iniciales de los LFSR, en forma secuencial.

3. Pruebas de aleatoriedad

3.1 Elección de las pruebas estadísticas

Fueron seleccionadas algunas pruebas de la Norma NIST Special Publication 800-

22, del trabajo de Rukhin (et al.) [10], estas son las siguientes: Prueba de frecuencia, Prueba de frecuencia en un bloque, Prueba de rachas, Prueba de rachas de unos en un bloque, Prueba de sumas acumuladas, Prueba de entropía aproximada

3.2 Pruebas sobre el generador

Se analizaron cien secuencias binarias de 100.000 bits, obtenidas del generador a partir de cien claves distintas. El nivel de significancia adoptado para las pruebas estadísticas es de $\alpha = 0{,}01$. La hipótesis nula es:

$$H_0 \rightarrow p\_valor > 0{,}01$$

3.3 Proporción de muestras que pasan las pruebas

Para el análisis de los resultados, se determina la proporción de muestras que superan las pruebas, y con esos datos se construye un gráfico de puntos, luego se verifica si los mismos caen dentro de los límites superior e inferior, donde k es el número de muestras. $LS, LI = (1-\alpha) \pm 3 \cdot \sqrt{\alpha \cdot (1-\alpha)/k}$

En nuestro caso $k = 100$ y el nivel de significancia elegido es: $\alpha = 0.01$, los límites quedan: $LS = 1{,}02$ y $LI = 0{,}96$

Se consideran todas pruebas, los resultados se indican en la tabla 3.

Tabla 3. Pruebas estadísticas

Pruebas	Proporción	Superior	Inferior
Frecuencias	1,00	1,02	0,96
Frecuencias en un Bloque	1,00	1,02	0,96
Rachas	1,00	1,02	0,96
Rachas de Unos en un Bloque	0,98	1,02	0,96
Sumas Acumuladas Adelante	1,00	1,02	0,96
Sumas Acumuladas Atrás	1,00	1,02	0,96
Entropía Aproximada	1,00	1,02	0,96

En la figura 2 se aprecia el resultado, en definitiva, las secuencias que entrega el generador superan las pruebas de aleatoriedad.

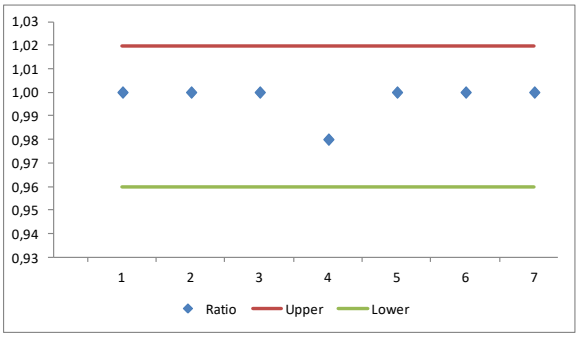


Fig. 2. Gráfico de puntos

4. Conversión de secuencia binaria a decimal

4.1 Procedimiento de conversión de secuencia binaria a secuencia decimal

Como la secuencia entregada por el generador es binaria y se necesita trabajar con números decimales, se adopta un criterio de conversión que es el siguiente:

- Dividir la secuencia binaria en bloques de 7 bits cada uno.
- Convertir esos bloques de 7 bits en número decimal.
- Como necesitamos números decimales comprendidos entre 0 y 99, los valores superiores a 99 se descartan.

Del resultado de la operación queda una secuencia decimal de 0 a 99 pseudoaleatoria. Finalmente queda verificar la aleatoriedad, se realiza una prueba de χ^2sobre la secuencia decimal obtenida.

4.2 Prueba de χ^2 sobre la secuencia decimal.

Realizamos la prueba de χ^2 sobre la secuencia decimal, y nos queda la siguiente tabla 4.

Tabla 4. Tabla de frecuencias y cálculo de χ^2

Clase *A*	Observada *B*	Esperada *C*	C-B *D*	D*D *E*	E/10 *F*
1	1000	1002	2	4	0,00
2	1000	1027	27	729	0,73
3	1000	966	-34	1156	1,16
4	1000	963	-37	1369	1,37
5	1000	1047	47	2209	2,21
6	1000	1002	2	4	0,00
7	1000	1036	36	1296	1,30
8	1000	980	-20	400	0,40
9	1000	1009	9	81	0,08
10	1000	968	-32	1024	1,02
	10000	**10000**			**8,27**

Finalmente comparamos el χ^2 de aceptación para un $\alpha = 0{,}01$ y 9 grados de libertad, con el χ^2 obtenido en tabla 4, de modo tal que resulte:

$$\boldsymbol{\chi^2 Aceptación > \chi^2 Obtenido}$$

$$\mathbf{21{,}66 > 8{,}27}$$

Cae en la zona de aceptación.

4.3 Generar números aleatorios para la primera y segunda variable

Se hace correr el generador de números aleatorios con la clave: 1X92ZQXu8f3oDzSEZRN7OcCM1LR97P9a y con los datos obtenidos, se realiza la tabla 5. Se repite el procedimiento con la clave: **P1kAlMiG2Kb7FzP5tM1QBI6DSS92c31A** y con los datos obtenidos, nos queda la tabla 6.

Tabla 5. Números aleatorios

67	82	30	21	97	91	47	19	94	11
89	94	14	58	44	17	74	21	13	75
…	…	…	…	…	…	…	…	…	…

Tabla 6. Números aleatorios

71	52	18	20	42	4	35	43	51	1
0	12	60	22	42	61	91	80	62	7
…	…	…	…	…	…	…	…	…	…

5. Simulación de Montecarlo

Se aplica el método de simulación de Montecarlo, cuando componentes del sistema son de comportamiento aleatorio [11]. El procedimiento tiene las siguientes etapas:

- Datos de las variables en estudio (observaciones).
- Confección Tabla de de frecuencias, probabilidades, probabilidades acumuladas e intervalos de números aleatorios.
- Generar los números aleatorios, con el generador presentado en este trabajo.
- Simular el experimento para un ciclo de 20 fallas.
- Realizar 200 repeticiones del experimento y obtener el promedio.

6. Reparación de máquinas

6.1 Procedimiento para simular dos variables

A continuación se realiza una simulación de Montecarlo donde se presenta dos variables aleatorias. En este caso se analiza un problema de reparación de máquinas, y se desarrolla un experimento de simulación para 20 fallas.

Datos de la primera variable.

En la tabla 7 figuran los datos del tiempo entre fallas de 200 observaciones.

Tabla 7. Tiempo entre fallas

3	4	1	3	2	2	2	3	6	4	4	5	6	2	6	1	4	1	1	2
4	6	3	5	5	1	4	4	5	3	2	4	4	2	4	5	5	3	1	4
5	1	4	1	6	2	6	2	6	4	4	4	6	3	1	2	6	4	1	2
3	5	5	1	1	1	1	4	1	4	3	3	3	5	1	6	4	5	6	2
4	2	2	2	1	4	6	2	1	1	1	1	4	1	2	6	3	4	3	5
4	6	1	3	6	2	5	6	4	1	5	4	6	1	6	6	5	6	4	6
4	4	3	2	2	1	4	6	6	3	3	5	2	6	1	5	4	6	2	6
2	4	2	4	6	5	5	5	4	1	5	2	2	6	6	4	2	6	1	6
3	3	1	6	2	2	2	2	5	4	1	5	6	4	2	5	4	1	1	3
2	5	4	5	2	6	3	3	2	3	2	6	1	2	5	5	5	5	5	2

Tabla de frecuencias, probabilidades, probabilidades acumuladas e intervalos de números aleatorios

En la tabla 8, aparece la distribución de frecuencia, las probabilidades, las probabilidades acumuladas e intervalos de números aleatorios del tiempo entre fallas.

Tabla 8. Intervalos de números aleatorios del tiempo entre fallas

Tiempo entre fallas	Frecuencia	Probabilidad	Probabilidad acumulada	Números aleatorios
1	34	0,17	0,17	0 al 16
2	38	0,19	0,36	17 al 35
3	23	0,11	0,47	36 al 46
4	39	0,20	0,67	47 al 66
5	31	0,15	0,82	67 al 81
6	35	0,18	1,00	82 al 99
	200	1,00		

Tiempos entre fallas aleatorios.

A partir de las tablas 5 y 8, en tabla 9 se establecen los tiempos entre fallas aleatorios.

Tabla 9. Tiempos entre fallas aleatorios

5	6	2	2	6	6	4	2	6	1
6	6	1	4	3	2	5	2	1	5
6	2	3	1	4	1	6	1	4	2
2	2	1	4	1	4	2	4	4	6
…	…	…	…	…	…	…	…	…	…

Datos de segunda variable.

En la tabla 10 figuran los tiempos de reparación para 200 observaciones.

Tabla 10. Tiempo de reparaciones

1	4	4	1	4	1	3	4	1	3	2	4	2	2	4	4	3	4	2	1
4	2	2	4	2	4	1	1	1	3	2	1	1	1	2	4	3	3	2	4
1	3	4	3	3	3	3	1	4	2	4	1	1	4	4	4	2	2	2	4
2	2	1	3	1	2	3	4	3	2	3	2	2	3	1	2	1	2	1	2
1	1	4	4	1	4	3	2	1	2	4	3	1	1	3	1	2	1	4	4
2	4	3	1	2	2	1	3	2	4	1	3	2	1	4	4	3	1	3	1
2	1	2	3	1	4	3	2	4	2	2	1	2	1	3	1	2	3	2	4
3	3	2	1	3	1	3	3	1	2	4	2	3	1	3	3	1	4	4	4
3	4	3	2	1	4	4	3	2	3	4	2	3	2	2	3	4	1	2	1
1	2	1	4	4	1	2	1	3	3	4	3	1	2	1	4	3	3	1	3

Tabla de frecuencias, probabilidades, probabilidades acumuladas e intervalos de números aleatorios

En la tabla 11, aparece la distribución de frecuencia, las probabilidades, las probabilidades acumuladas e intervalos de números aleatorios del tiempo de reparaciones.

Tabla 11. Intervalos de números aleatorios

Tiempo de reparación	Frecuencia	Probabilidad	Probabilidad acumulada	Números aleatorios
1	54	0,27	0,27	0 al 26
2	51	0,26	0,53	27 al 52
3	48	0,24	0,77	53 al 76
4	47	0,23	1,00	77 al 99
	200	1,00		

Tiempos de reparación aleatorios.

A partir de las tablas 6 y 11, en tabla 12 se establecen los tiempos de reparaciones aleatorios.

Tabla 12. Tiempo de reparaciones aleatorios

3	2	1	1	2	1	2	2	2	1
1	1	3	1	2	3	4	4	3	1
3	4	2	3	4	3	4	3	1	2
4	4	1	3	4	2	1	2	2	4
…	…	…	…	…	…	…	…	…	…

Simular el experimento.

De la tabla 5 y 6 de números aleatorios se seleccionan 20 números de cada una de ellas, para realizar el experimento en un ciclo de 20 fallas, entonces nos queda la tabla 13.

Tabla 13. Experimento para determinar el tiempo inactivo de las máquinas

Número de fallas	Aleatorio tabla 5	Tiempo entre fallas	Inicio de falla	Inicio reparación	Aleatorio de tabla6	Tiempo de reparación	Fin de reparación	**Tiempo inactivo**	Espera
1	67	5	5	5	71	3	8	**3**	0
2	82	6	11	11	52	2	13	**2**	0
3	30	2	13	15	18	1	14	**1**	0
4	21	2	15	17	20	1	16	**1**	0
5	97	6	21	21	42	2	23	**2**	0
6	91	6	27	27	4	1	28	**1**	0
7	47	4	31	31	35	2	33	**2**	0
8	19	2	33	33	43	2	35	**2**	0
9	94	6	39	39	51	2	41	**2**	0
10	11	1	40	43	1	1	42	**2**	1
11	89	6	46	46	0	1	47	**1**	0
12	94	6	52	52	12	1	53	**1**	0
13	14	1	53	56	60	3	56	**3**	0
14	58	4	57	57	22	1	58	**1**	0
15	44	3	60	60	42	2	62	**2**	0

16	17	2	62	62	61	3	65	**3**	0
17	74	5	67	67	91	4	71	**4**	0
18	21	2	69	70	80	4	75	**6**	2
19	13	1	70	71	62	3	78	**8**	5
20	75	5	75	75	7	1	79	**4**	3
							79	**51**	

Tiempo inactivo de las máquinas: **51 horas** en un ciclo de 20 fallas.

Realizar varias simulaciones y obtener el resultado.

Para obtener un mejor resultado realizamos muchas simulaciones del experimento anterior, en nuestro caso adoptamos 200 simulaciones y calculando el promedio con los resultados obtenidos de las simulaciones, que figura en la tabla 14:

Tabla 14**.** Simulaciones de tiempo inactivo de las máquinas

51	131	60	58	44	53	88	66	132	65	85	70	64	74	65	47	54	48	60	43
71	51	70	55	53	87	77	62	49	61	68	54	48	53	99	44	75	48	74	63
64	46	101	58	52	62	63	65	70	50	43	57	101	55	69	63	90	67	61	58
68	65	52	68	48	55	84	53	74	53	68	62	57	43	58	57	69	72	62	54
54	60	54	101	52	94	111	74	127	65	59	47	55	68	58	56	90	55	110	60
68	64	105	45	63	85	64	65	76	60	44	60	63	90	77	53	55	46	65	65
65	57	69	84	63	72	71	55	60	73	117	58	77	69	85	51	58	69	66	54
64	51	56	56	69	85	83	100	46	60	60	115	68	59	54	69	75	56	59	49
89	55	71	61	93	50	61	69	55	78	57	53	60	54	48	53	51	66	62	82
57	91	71	63	73	54	99	65	67	48	43	77	45	68	49	60	67	54	51	71

El promedio de las 200 simulaciones, indicadas en la tabla 14, es de **65,74 horas** de tiempo inactivo de las máquinas, en el ciclo estudiado de 20 fallas.

7. Conclusiones

Uno de los usos de los generadores de números aleatorios es la realización de las simulaciones de Montecarlo, donde se requieren variables aleatorias para el cálculo de los experimentos. Existen muchos casos de simulaciones donde aparecen una, dos, tres o múltiples variables aleatorias.

En este trabajo aparecen dos variables y se generaron dos tablas aleatorias a partir del generador presentado. Con esos valores se realizaron las 200 simulaciones que determinaron el promedio del tiempo inactivo de las máquinas para un ciclo de 20 fallas.

El uso del generador permite, con respecto al cálculo mediante planillas de números aleatorios, un proceso más rápido y con mayor cantidad de simulaciones.

Referencias

1. Massodi, F., Alam, S. and Bokhari, M., "A Analysis of Linear Feedback Shift Registers in Stream Ciphers", International Journal of Computer Application, 16 (17), pp. 0975 – 887, 2012.
2. Menezes, A.,Van Oorschot, P. and Vanstone, S., "Handbook of Applied Cryptography", Massachusetts Institute of Technology, 1996.
3. Parr, C. and Pelzl, L., Understanding Cryptography, Springer, 2010.
4. Stahnke, W., "Primitive Binary Polynomials", Mathematics of Computation, 27. 124, pp. 977-980, 1973.
5. Seroussi, G., "Table of Low-Weight Binary Irreducible Polynomials", Computer Systems Laboratory, 1998.
6. Clark, J., Jacob, J., Maitra, S., Stanica, P.: Almost Boolean Functions: The Design of Boolean Functions by Spectral Inversion. Computational intelligence. 20. (3), 450—462 (2004)
7. Braeken, A.: Cryptographic Properties of Boolean Functions and S-Boxes. Faculteit Ingenieurswetenschappen. Katholieke Universiteit Leuven (2003)
8. Elhosary, A., Hamdy, N., Farag, I., Rohiem, I.: State of the Art in Boolean Functions Cryptographic Assessment. International Journal of Computer Networks and Communications Security.1. (3), 88--94 (2013)
9. Fishman, G.: Multiplicative Congruential Random Number Generators with Modulus 2ß : An Exhaustive Analysis for ß = 32 and a Partial Analysis for ß = 48. Mathematics of Computation. 54. (189), 33--344 (1990)
10. Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S., Levenson, M., Vangel, M., Banks, D., Heckert, A., Dray, J., and Vo, S., "A Statistical Prueba Suite for Random and Pseudorandom Number Generators for Cryptographic Applications", National Institute of Standards and Technology, (2000).
11. Heizer, J. y Render, B. Dirección de la producción y de operaciones. Decisiones tácticas, 8.ª edición PEARSON EDUCACIÓN, S.A., Madrid, (2008)

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Generador Binario Pseudoaleatorio Basado en la Combinación de Registros de Desplazamiento de Retroalimentación Lineal con Función de Filtrado No Lineal

Andrés Francisco Farías – Andrés Alejandro Farías

Departamento Académico de Ciencias Físicas, Matemáticas y Naturales
Universidad Nacional de La Rioja, La Rioja. Argentina
afarias665@yahoo.com.ar, andres_af86@hotmail.com

Abstract. Este trabajo desarrolla el procedimiento para la construcción de un generador binario pseudoaleatorio basado en la combinación mediante función por mayoría de registros de desplazamiento con retroalimentación no lineal (Linear Feedback Shift Register, LFSR).

El proceso incluye: características de los LFSR, definición del modelo, elección de los distintos LFSR, selección de funciones booleanas en base a sus óptimas propiedades criptográficas, composición del generador con los componentes seleccionados, clave y el procedimiento para generarlas, elección de las pruebas estadísticas a utilizar y los criterios de análisis de los resultados, puesta en funcionamiento y realización de las pruebas de aleatoriedad necesarias sobre las secuencias obtenidas.

Keywords: LFSR, generator, key, boolean function, non-linearity.

1. Introducción

Los generadores binarios pseudoaleatorios deben entregar secuencias de calidad y además se les exige imprevisibilidad y facilidad de implementación, pero, fundamentalmente un período con una longitud significativa.

Es en esos términos que se propone un modelo que responda a tales exigencias. La modalidad elegida se basa en la combinación no lineal de generadores conformados por ocho NLFSR, que tienen una función booleana de filtrado [1], [2].

El procedimiento de construcción de un generador pseudoaleatorio de ese estilo requiere de varias etapas:

- Características de los LFSR
- Definición del modelo.
- Elección de los distintos LFSR, base del sistema.
- Selección de funciones booleanas de cuatro variables en base a sus óptimas propiedades criptográficas.
- Composición del generador con los componentes ya seleccionados.
- Clave y el procedimiento para generar los estados iniciales de los LFSR.
- Elección de las pruebas estadísticas a utilizar y los criterios de análisis de los resultados.

- Puesta en funcionamiento y realización de las pruebas de aleatoriedad necesarias sobre las secuencias obtenidas. Generador binario pseudoaleatorio

1.1 Características de los LFSR

Los LFSR que se utilizan tienen la siguiente estructura:

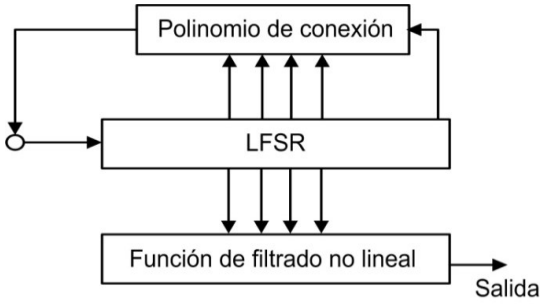


Fig. 1. Esquema del LFSR

1.2 Definición esquemática del modelo

El generador propuesto en este trabajo, está conformado por cuatro LFSR, que tienen cada uno, una función booleana de filtrado no lineal, que producen secuencias binarias, las que luego son combinadas mediante otras funciones booleanas de combinación, y una operación final XOR, según la figura 2:

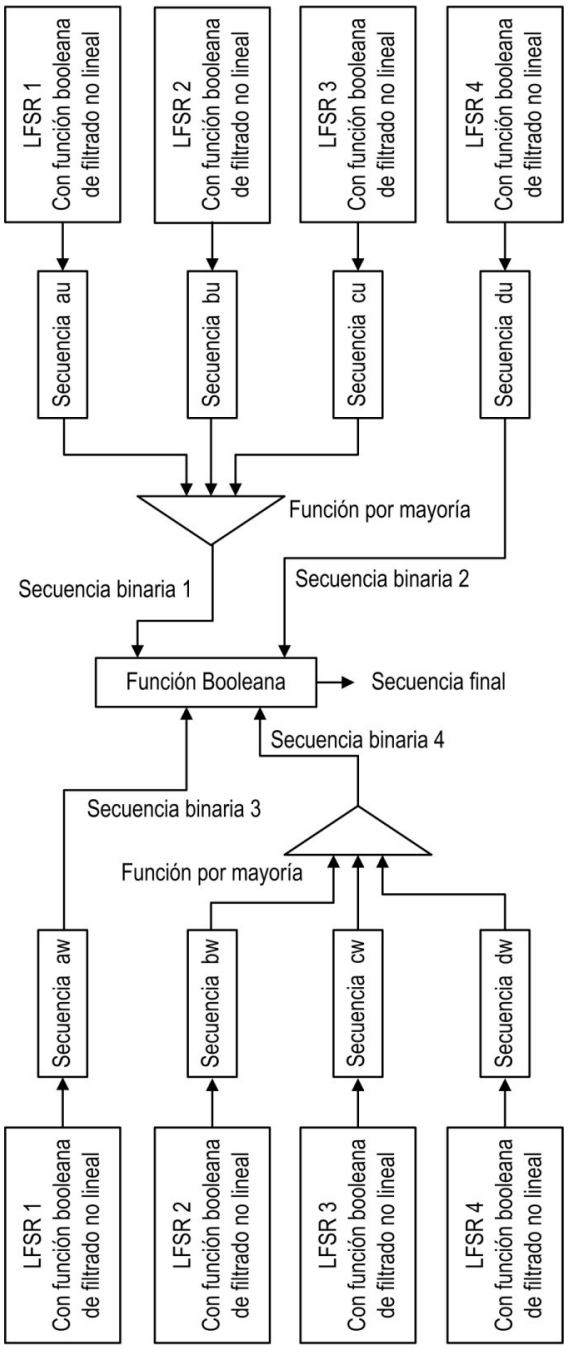


Fig. 2. Esquema generador binario pseudoaleatorio

1.3 Elección de los distintos LFSR

Las longitudes y polinomios primitivos de cada LFSR, que componen el generador, son las siguientes [3], [4], [5], en la tabla 1 y en la figura 2 se indican como queda el dispositivo.

Tabla 7. LFSR, longitudes y polinomios primitivos del Generador

LFSR	Longitud	Polinomios primitivos
2	41	$P(x)_3 = x^{41} + x^{32} + x^{31} + x^{27} + 1$
3	43	$P(x)_4 = x^{43} + x^{27} + x^{22} + x^5 + 1$
3	59	$P(x) = x^{59} + x^{54} + x^{46} + x^{26} + 1$
4	53	$P(x) = x^{53} + x^{50} + x^{41} + x^{20} + 1$

1.4 Selección de las funciones booleanas

Propiedades criptográficas deseables.

A continuación, se indican algunas de las propiedades criptográficamente más significativas, adoptadas para este trabajo [6], [7], [8].

- **Función Balanceada:**
- **No Linealidad**:
- **Grado Algebraico**:
- **Criterio de Avalancha Estricto**

Siguiendo los criterios arriba indicados las funciones booleanas aceptadas, son observadas en tabla 2:

Taba 8. Funciones de cuatro variables adoptadas

f_{NAF}	No linealidad	Cumple SAC
$f_{84} = a \cdot c \oplus b \cdot c \oplus a \cdot d \oplus b \cdot d \oplus c \cdot d$	4	Sí
$f_{89} = a \cdot c \oplus b \cdot c \oplus d \oplus a \cdot d \oplus b \cdot d$	4	Sí
$f_{100} = a \cdot c \oplus b \cdot c \oplus d \oplus a \cdot b \cdot d \oplus c \cdot d$	4	Sí
$f_{176} = c \oplus a \cdot c \oplus b \cdot c \oplus a \cdot d \oplus b \cdot d$	4	Sí
$f_{199} = c \oplus a \cdot c \oplus b \cdot c \oplus d \oplus c \cdot d$	4	Sí
$f_{381} = c \oplus a \cdot b \cdot c \oplus a \cdot d \oplus b \cdot d \oplus c \cdot d$	4	Sí
$f_{468} = c \oplus d \oplus a \cdot d \oplus b \cdot d \oplus c \cdot d$	4	Sí
$f_{536} = a \cdot b \oplus b \cdot c \oplus a \cdot d \oplus b \cdot d \oplus c \cdot d$	4	Sí
$f_{5056} = a \oplus b \oplus a \cdot d \oplus b \cdot d \oplus c \cdot d$	4	Sí

1.5 Conformación del generador combinacional

El generador combinacional se indica en la figura 3:

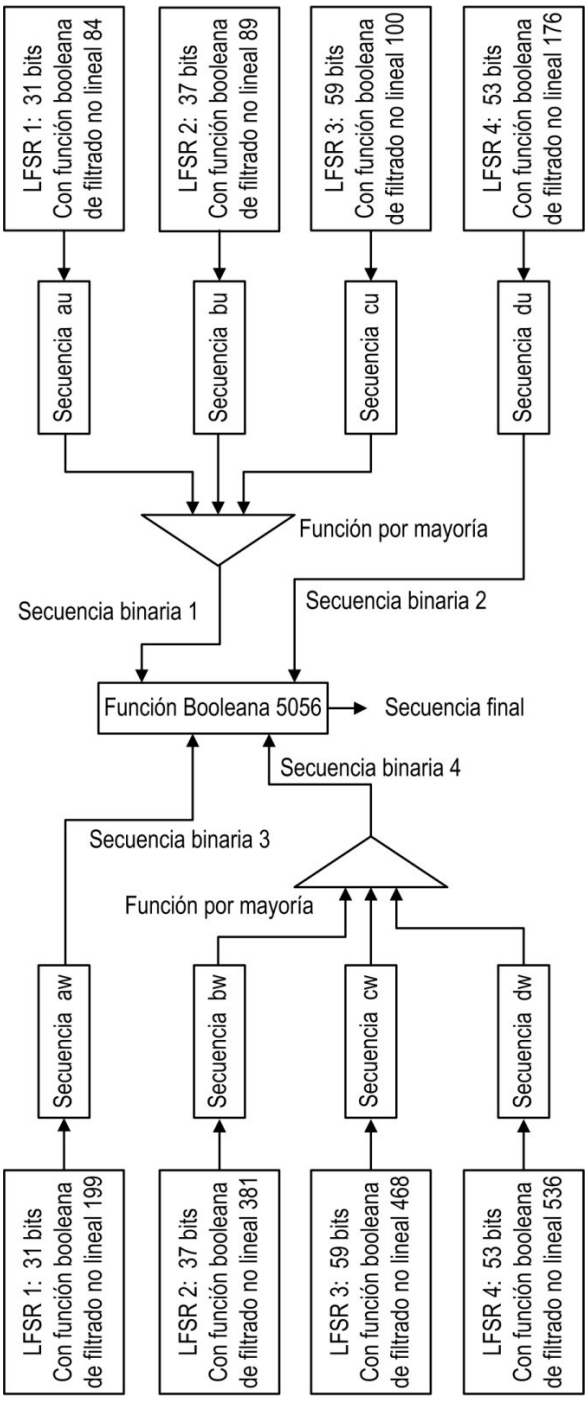


Fig. 3. Generador Combinacional

1.6 Clave

Para originar los estados iniciales de los distintos LFSR se realiza un proceso que utiliza una clave de 32 caracteres, que expresada en código ASCII (American Standard Code for Information Interchange), tiene longitud de 256 bits.

La clave es sometida a un proceso criptográfico, que se indica en la figura 4.

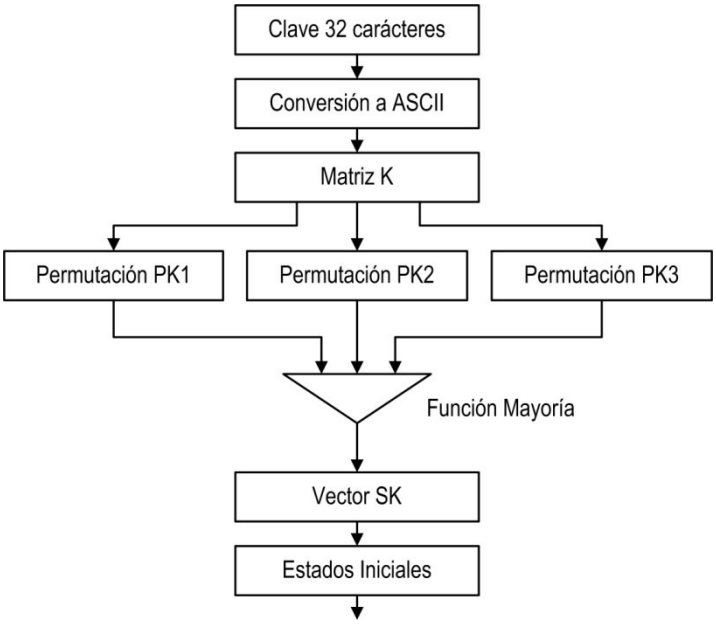


Fig. 4. Clave para el generador

1.7 Permutaciones

Generador congruencial multiplicativo: El generador tiene la siguiente expresión: [9]

$$x_{i+1} = (a_x \cdot x_i)mod\ m_x \qquad (4)$$

Donde:

$$a_x = multiplicador$$
$$m_x = módulo$$
$$x_0 = semilla$$

Tabla.3. Vectores, módulos, multiplicadores y semillas

Vector	módulo	multiplicador	semilla
PK1	1048576	1747	3249
PK2	1048576	1753	3271
PK3	1048576	1759	3301

De la operación resulta un vector SK[j] de 256 bits, que es el que proveerá los estados iniciales de los LFSR, en forma secuencial.

2. Pruebas de aleatoriedad

2.1 Elección de las pruebas estadísticas

Fueron seleccionadas algunas pruebas de la Norma NIST Special Publication 800-22, del trabajo de Rukhin (et al.) [10].

- Prueba de frecuencia
- Prueba de frecuencia en un bloque
- Prueba de rachas
- Prueba de rachas de unos en un bloque
- Prueba de sumas acumuladas
- Prueba de entropía aproximada

2.2 Pruebas sobre el generador

Se analizaron cien secuencias binarias de 100.000 bits, obtenidas del generador a partir de cien claves distintas. El nivel de significancia adoptado para las pruebas estadísticas es de $\alpha = 0{,}01$. La hipótesis nula es:

$$H_0 \rightarrow \text{p\_valor} > 0{,}01$$

2.3 Proporción de muestras que pasan las pruebas

Para el análisis de los resultados, se determina la proporción de muestras que superan las pruebas, y con esos datos se construye un gráfico de puntos, luego se verifica si los mismos caen dentro de los límites superior e inferior, donde k es el número de muestras. $LS, LI = (1-\alpha) \pm 3 \cdot \sqrt{\alpha \cdot (1-\alpha)/k}$

En nuestro caso $k = 100$ y el nivel de significancia elegido es: $\alpha = 0.01$, los límites quedan: $LS = 1{,}02$ y $LI = 0{,}96$

Se consideran todas pruebas, los resultados se indican en la tabla 4.

Table 9. Pruebas estadísticas

Pruebas	Proporción	Superior	Inferior
Frecuencias	1,00	1,02	0,96
Frecuencias en un Bloque	0,97	1,02	0,96
Rachas	1,00	1,02	0,96
Rachas de Unos en un Bloque	0,98	1,02	0,96
Sumas Acumuladas Adelante	1,00	1,02	0,96
Sumas Acumuladas Atrás	0,97	1,02	0,96
Entropía Aproximada	1,00	1,02	0,96

En la figura 5 se aprecia el resultado, en definitiva, las secuencias que entrega el generador superan las pruebas de aleatoriedad.

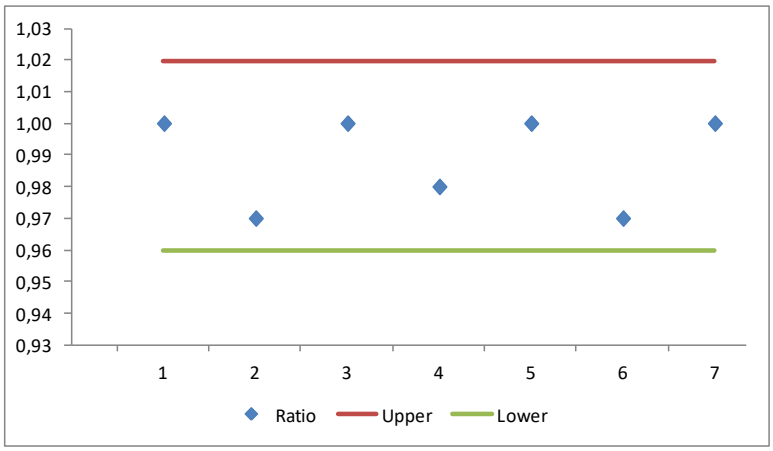


Fig. 5. Gráfico de puntos

3. Conclusiones

Se diseñó un dispositivo que combina en forma no lineal las secuencias producidas por ocho generadores compuestos por LFSR con funciones paralelas de filtrado. Para se dispuso del auxilio de la función por mayoría, junto con una función booleana final de cuatro variables.

Los NLFSR que componen cada generador tienen polinomios de conexión primitivos, lo que asegura un elevado período en la secuencia resultante.

Las funciones booleanas son las responsables del proceso no lineal y aseguran las mejores prestaciones criptográficas. Realizado el proceso de selección, las funciones fueron incorporadas al generador y luego puestas a funcionar para generar las secuencias respectivas con distintos valores de claves y ser sometidas a las pruebas de aleatoriedad.

Los resultados obtenidos fueron satisfactorios, por lo que el modelo presentado se considera válido para la generación de secuencias pseudoaleatorias de buena calidad criptográfica.

Referencias

1. Massodi, F., Alam, S. and Bokhari, M., “A Analysis of Linear Feedback Shift Registers in Stream Ciphers”, International Journal of Computer Application, 16 (17), pp. 0975 – 887, 2012.
2. Menezes, A.,Van Oorschot, P. and Vanstone, S., “Handbook of Applied Cryptography”, Massachusetts Institute of Technology, 1996.
3. Parr, C. and Pelzl, L., Understanding Cryptography, Springer, 2010.

4. Stahnke, W., "Primitive Binary Polynomials", Mathematics of Computation, 27. 124, pp. 977-980, 1973.
5. Seroussi, G., "Table of Low-Weight Binary Irreducible Polynomials", Computer Systems Laboratory, 1998.
6. Clark, J., Jacob, J., Maitra, S., Stanica, P.: Almost Boolean Functions: The Design of Boolean Functions by Spectral Inversion. Computational intelligence. 20. (3), 450—462 (2004)
7. Braeken, A.: Cryptographic Properties of Boolean Functions and S-Boxes. Faculteit Ingenieurswetenschappen. Katholieke Universiteit Leuven (2003)
8. Elhosary, A., Hamdy, N., Farag, I., Rohiem, I.: State of the Art in Boolean Functions Cryptographic Assessment. International Journal of Computer Networks and Communications Security.1. (3), 88--94 (2013)
9. Fishman, G.: Multiplicative Congruential Random Number Generators with Modulus 2ß : An Exhaustive Analysis for ß = 32 and a Partial Analysis for ß = 48. Mathematics of Computation. 54. (189), 33--344 (1990)
10. Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S., Levenson, M., Vangel, M., Banks, D., Heckert, A., Dray, J., and Vo, S., "A Statistical Prueba Suite for Random and Pseudorandom Number Generators for Cryptographic Applications", National Institute of Standards and Technology, (2000).

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Generación De Nombres De Dominios Mediante Redes Neuronales Híbridas CNN-LSTM

Reynier Leyva La O[1,2], Rodrigo Gonzalez[1], and Carlos A. Catania[2]

GridTICs, Facultad Regional Mendoza,
Universidad Tecnológica Nacional, Mendoza, Argentina
rleyvalao@mendoza-conicet.gob.ar
LABSIN, Facultad de Ingeniería. Universidad Nacional de Cuyo, Mendoza, Argentina
harpomaxx@gmail.com

Resumen El uso creciente de algoritmos de generación de dominios (DGA) para la comunicación entre servidores de comando y control (C&C) plantea serios desafíos en el proceso de detección de botnets. Los especialistas en seguridad deben estar en permanente actualización de las nuevas técnicas a fin de poder realizar una detección eficiente. En este trabajo, se propone el uso de una red neuronal que combina las técnicas de Convolutional Neural Network (CNN) y Long Short-Term Memory (LSTM) para generar nombres de dominios de manera algorítmica. El objetivo de contar con un generador de nombres de dominio es la de poder utilizar la información provista por el mismo en el proceso de detección de los mismos. Por lo que para su validación se realizaron pruebas a los dominios generados utilizando un detector de DGA y se encontró que los nombres generados no fueron fácilmente identificados como generados por DGA. Dicho detector mostró una exactitud de un 10%. Estos resultados sugieren que el generador de nombres propuesto puede ser una herramienta efectiva para detectar limitaciones en las técnicas de detección de DGA.

Palabras Claves DGA, seguridad informática, CNN, LSTM, nombres de dominio.

1. Introducción

Un DGA (Domain Generation Algorithm) es un algoritmo utilizado por los ciberdelincuentes para generar nombres de dominio de forma automática y aleatoria. Su principal objetivo es establecer una comunicación segura y encubierta entre los bots y el servidor de comando y control (C&C). A diferencia de los dominios estáticos y predefinidos, que son más fáciles de detectar y bloquear, los dominios generados por DGA cambian constantemente, lo que dificulta su detección y bloqueo por parte de los sistemas de seguridad. Los DGA se utilizan comúnmente en las botnets para evitar ser detectados y para garantizar una mayor resiliencia y persistencia en las comunicaciones [1].

El funcionamiento de un DGA se basa en una semilla inicial o clave, a partir de la cual se generan secuencias de caracteres que formarán los nombres de dominio. Estas

secuencias pueden variar en longitud y estructura, y se suelen utilizar técnicas de cifrado y ocultamiento para dificultar aún más su detección.

Al generar constantemente nuevos nombres de dominio, los ciberdelincuentes pueden mantener su infraestructura activa y operativa, lo que les permite controlar los bots y llevar a cabo sus actividades maliciosas sin ser detectados fácilmente.

Los DGA representan un desafío significativo para la seguridad cibernética debido a su capacidad para evadir las medidas de detección y bloqueo tradicionales.

En particular, el presente trabajo fue motivado por dos aspectos fundamentales: fortalecer la capacidad de los detectores de Dominios Generados Algorítmicamente (AGD) y explorar el potencial del aprendizaje profundo (deep learning) en la generación de nombres de dominios. Para tal fin se entrenó una red neuronal utilizando técnicas de aprendizaje supervisado. La red consta de una capa convolucional CNN que se encarga de extraer características relevantes de los nombres de dominio. Luego una capa LSTM ayuda a capturar patrones secuenciales y contextuales en la generación de los dominios. Posteriormente, se evalúan los dominios generados utilizando varios detectores de AGD existentes para ver si el modelo creado genera nombres de dominios difíciles de distinguir por los detectores. Entonces estos dominios pueden ser utilizados como casos de prueba para mejorar y fortalecer los detectores de AGD, ayudando a identificar y corregir posibles vulnerabilidades en su funcionamiento.

Es importante destacar que la generación de dominios poco distinguibles ofrece beneficios significativos en términos de fortalecimiento de los detectores de AGD y mejora de la seguridad cibernética. Por un lado, permite evaluar la eficacia de los detectores existentes y descubrir posibles vulnerabilidades, lo que impulsa su mejora continua. Además, la capacidad de generar dominios que se asemejan a los dominios legítimos aumenta la capacidad de detección y bloqueo de comunicaciones maliciosas. Sin embargo, es importante tener en cuenta las limitaciones, como el posible mal uso de esta tecnología, la necesidad de adaptar constantemente los detectores de AGD ante nuevas técnicas de generación de dominios y la posibilidad de falsos positivos y negativos que podrían afectar la precisión de los sistemas de detección.

En resumen, las contribuciones del presente estudio son por un lado explorar la factibilidad de utilizar redes neuronales en la generación de dominios y su evaluación por parte de detectores.

2. Metodología

En esta sección, se aborda el enfoque metodológico utilizado para generar dominios similares a los dominios normales. El objetivo perseguido consistió en concebir una red neuronal híbrida que combinara capas convolucionales (CNN) y memoria a largo plazo (LSTM) para generar nombres de dominio difíciles de distinguir de los dominios legítimos. Para lograrlo, se procedió a procesar los datos de dominios normales y a vectorizarlos para entrenar la red neuronal, para esto se utilizó Keras, un framework de aprendizaje profundo. Esta combinación permitió la generación de nombres de dominio que se asemejaban a los normales. A través de este proceso, se buscó fortalecer la detección de AGD y mejorar la seguridad cibernética en general. Se utilizó una amplia cantidad de dominios normales para un entrenamiento óptimo. Este enfoque

proporciona una base sólida para mejorar la detección de AGD y fortalecer la seguridad contra posibles ataques de botmasters.

2.1 Descripción del Conjunto de Datos Utilizado

Para este trabajo se utilizó un dataset que contenía tanto AGD como dominios normales. Los dominios normales fueron extraídos de los 1 millón de dominios más populares según Alexa [24], y se incluyeron adicionalmente 3.161 dominios normales proporcionados por el feed de Bambenek Consulting [23]. Este último grupo de dominios es particularmente interesante, ya que consiste en dominios sospechosos que no fueron generados por DGA. En total, la base de datos contenía 1.003.161 dominios normales [19].

Por otro lado, los dominios generados por DGA se obtuvieron de los repositorios de dominios AGD de Andrey Abakumov [22] y John Bambenek. En total, había 1.915.335 dominios DGA que correspondían a 51 familias diferentes de malware. Cabe destacar que aproximadamente el 55% de la porción DGA del dataset estaba compuesta por muestras de las familias de malware

Banjori, Post, Timba, Cryptolocker, Ramdo y Conficker.

En el proceso de generación de dominios, se optó por utilizar únicamente los dominios normales de este dataset, ya que el objetivo era entrenar la red neuronal para generar nombres de dominio similares a los dominios normales.

Con base en el conjunto de datos descrito anteriormente, en la subsección 2.2, se aborda detalladamente el proceso de preprocesamiento de los datos. Este paso es fundamental para preparar los dominios normales como entrada para la red neuronal y asegurar que la generación de nombres de dominio se realice de manera eficiente y efectiva.

2.2 Preprocesamiento de los Datos

Es esencial resaltar la importancia del preprocesamiento de los dominios como paso fundamental para prepararlos adecuadamente antes de introducirlos en la red neuronal. Este proceso de preparación de los datos juega un papel crucial en asegurar que la red pueda comprender y procesar la información de manera efectiva. Al realizar el preprocesamiento, los dominios son transformados en formatos adecuados y estructurados, lo que facilita su manejo por parte de la red neuronal. Además, durante esta etapa, se llevan a cabo transformaciones y codificaciones necesarias para representar los dominios en forma de secuencias vectorizadas, permitiendo que la red neuronal aprenda las relaciones entre los caracteres y sus posiciones dentro de la secuencia de entrada. Mediante un preprocesamiento adecuado, se logra maximizar la calidad del entrenamiento y la capacidad de la red para generar nombres de dominio similares a los dominios normales. Esto resulta esencial para el cumplimiento del objetivo principal del estudio.

Para llevar a cabo el preprocesamiento de los dominios, se completó cada dominio para que todos tuvieran una longitud específica. Esto se logró mediante la adición de caracteres especiales (espacios) al final de cada dominio, siempre y cuando la longitud

del dominio fuera menor que la deseada. De esta manera, todos los dominios alcanzaron la misma longitud deseada. Luego, los dominios completados se unieron en una sola cadena de texto, utilizando el carácter espacio como separador entre cada dominio. Este proceso de preprocesamiento garantizó que los dominios tuvieran una estructura uniforme.

Una vez obtenida la cadena de texto completa, se procedió a dividirla en secuencias más pequeñas con un desplazamiento de 3 caracteres en cada iteración. De esta manera, se generaron las secuencias de entrada (sec X) junto con sus correspondientes caracteres siguientes (car y). Un ejemplo básico para entender cómo estaban compuestas las secuencias de entrada (sec X) y el carácter de salida (car y) se puede visualizar en la tabla 1.

Tabla 1. Secuencia de entrada y carácter de salida

No. Secuencia	Entrada	Carácter Salida
1	www.google.com	“ ”
2	www.google.	“c”
3	www.goog	“l”

Una vez obtenidas las secuencias de entrada y el carácter de salida, se procedió a aplicar tokenización a dichos datos. El proceso de tokenización es un componente fundamental en el preprocesamiento de datos para la generación de dominios. Al convertir cada carácter único en un token, se logra una representación más compacta y procesable para la red neuronal. Esto reduce significativamente la complejidad computacional del modelo y acelera el proceso de entrenamiento, ya que la red solo necesita aprender los patrones de los tokens en lugar de lidiar con cada carácter individual. Además, la tokenización permite una mejor generalización del modelo, ya que la red puede aprender a reconocer y manipular patrones comunes de caracteres, en lugar de memorizar cada carácter específico presente en los datos de entrenamiento.

En el contexto de este trabajo, una vez obtenidos los datos de entrada y salida, se llevó a cabo la identificación de los caracteres únicos presentes en las secuencias, formando así un alfabeto. A cada carácter se le asignó un índice correspondiente en dicho alfabeto. Para representar los caracteres de manera procesable por la red neuronal, se implementó la codificación one-hot [11], [8]. Mediante esta codificación, cada carácter se transformó en un vector disperso donde todas las posiciones eran 0, excepto una que contenía el valor 1 en la posición correspondiente al índice del carácter en el alfabeto. De esta forma, cada carácter se convirtió en un vector único y procesable por la red, lo que permitió que la red neuronal aprendiera eficientemente las relaciones entre los caracteres en las secuencias de entrada.

Este enfoque de codificación one-hot posibilitó la creación de un conjunto de datos adecuado para el entrenamiento y ajuste de la red neuronal, proporcionando la información necesaria para predecir los caracteres siguientes en función de las secuencias de entrada. Asimismo, la codificación one-hot permitió que la red comprendiera y procesara los caracteres de manera efectiva. Este proceso de tokenización y codificación fue fundamental para la eficiencia del modelo en la generación de dominios.

A continuación, se detallará la red neuronal utilizada en la arquitectura para la generación de dominios similares, en la subsección 2.3.

2.3 Descripción de la Red Neuronal

Como se mencionó en la subsección 2.2, en esta subsección se describirá en detalle la red neuronal utilizada para la generación de dominios similares a los normales.

En el proceso de selección de las redes neuronales para la arquitectura, se tuvo en cuenta la naturaleza secuencial de los datos y la necesidad de capturar características importantes de los dominios normales. Para lograr este objetivo, se optó por emplear capas de redes neuronales convolucionales (CNN) y redes de memoria de corto-largo plazo (LSTM).

Las redes neuronales convolucionales son ampliamente conocidas por su capacidad para extraer características relevantes de los datos de entrada, especialmente en imágenes y secuencias [11], [7]. En el contexto de generación de dominios, estas capas fueron fundamentales para capturar patrones y características clave presentes en los dominios normales. Permitieron que la red aprendiera las estructuras esenciales necesarias para generar dominios similares y coherentes.

Por otro lado, redes de memoria de corto-largo plazo (LSTM) son un tipo de red neuronal recurrente (RNN) diseñada para trabajar con secuencias de datos [12]. Estas capas resultaron esenciales para capturar dependencias a largo plazo en los datos de entrada, un aspecto crítico para la generación de secuencias de texto coherentes y realistas. La inclusión de LSTM en la arquitectura permitió que la red aprendiera las relaciones temporales y las dependencias complejas entre los caracteres en las secuencias de entrada.

La combinación de capas de redes neuronales convolucionales y capas LSTM en una sola red neuronal híbrida resultó en una arquitectura efectiva para la generación de dominios similares a los dominios normales. Cada capa desempeñó un papel importante en el proceso, permitiendo que la red aprendiera y generara secuencias de texto de manera coherente y realista. Estas características contribuyeron significativamente al logro del objetivo principal del estudio. La arquitectura en detalle seleccionada fue la siguiente:

1. **Capa de entrada:** Se definió una capa de entrada con la forma correspondiente a la longitud máxima de las secuencias y el número de caracteres únicos en los datos.
2. **Capa convolucional 1D:** Se agregó una capa convolucional con 128 filtros y un tamaño de kernel de 3, con una función de activación 'relu'. Esta capa ayudó a extraer características relevantes de las secuencias de entrada.
3. **Capa de agrupación máxima 1D:** Se añadió una capa de agrupación máxima con un tamaño de agrupación de 2. Esta capa permitió reducir la dimensionalidad de las características extraídas por la capa convolucional.
4. **Capa LSTM:** Se incluyó una capa LSTM con 128 unidades. La capa LSTM permitió capturar las dependencias temporales en las secuencias de entrada.
5. **Capa densa:** Se agregó una capa densa con una función de activación 'softmax' para la clasificación de los caracteres. El número de unidades en esta capa corresponde al número de caracteres únicos en los datos.

Con los tipos de redes neuronales previamente definidos, el enfoque se dirige ahora hacia la implementación y entrenamiento de la arquitectura, haciendo uso de la biblioteca Keras. Esta herramienta proporciona las funcionalidades necesarias para aprovechar al máximo la capacidad de generación de texto. En la siguiente subsección

(2.4), se aborda en detalle la utilización de Keras y cómo esta biblioteca facilita concretamente la generación de nombres de dominio. Este aspecto es fundamental para comprender exhaustivamente el proceso de generación de dominios y para obtener una comprensión completa de los resultados obtenidos en este estudio.

2.4 Algoritmo para generación de Dominios

En la presente subsección, se expone la implementación del código empleado en la generación de dominios mediante el empleo de Keras. En esta exposición, se abordarán minuciosamente los pormenores concernientes a la arquitectura de la red neuronal híbrida, así como las configuraciones y parámetros particulares que fueron empleados para el adiestramiento del modelo. El enfoque primordial se centra en la manera en que la biblioteca Keras fue utilizada con el propósito de materializar la generación de dominios que presentaran similitud con aquellos considerados como normales.

Para el entrenamiento del modelo, se utilizó el optimizador **Adam** [20] con una tasa de aprendizaje de 0.01. El modelo fue compilado utilizando la función de pérdida **categorical crossentropy** [21]. Con esta estructura de modelo, se buscó que la red neuronal aprendiera a generar nombres de dominios similares a los dominios normales de manera efectiva, contribuyendo así al objetivo principal del estudio.

Para obtener más detalles sobre la implementación y creación del modelo, el código se encuentra disponible en el repositorio público de GitHub [18]. En la porción de código presentada en la Figura 1, se exhibe la sección pertinente del código que corresponde a la creación del modelo:

Durante el proceso de entrenamiento de la red neuronal, se realizaron varias pruebas con distintos valores de épocas, tamaño de batch, diversidad, cantidad de neuronas por capa, y métodos de optimización, entre otros hiperparámetros. Sin embargo, la configuración de modelo que se mencionó previamente demostró ser la más efectiva. Después de cada época, se generaba nuevo texto para evaluar el rendimiento del modelo en la generación de nombres de dominio. Esta estructura permitió que la red neuronal generara de manera consistente y realista nombres de dominios. Para obtener información más detallada acerca de la implementación del entrenamiento del modelo descrito en la Figura 1, se proporciona el código correspondiente al proceso de entrenamiento en la Figura 2:

En cada época se elegía aleatoriamente una secuencia inicial de la cadena de texto y se utilizaba un valor de diversidad de 0.5. A partir de esta secuencia inicial, la red neuronal previamente entrenada generaba caracteres adicionales para completar la secuencia de texto. Se generaban específicamente 400 caracteres en los cuales se encontraban 9 nombres de dominios. Estos dominios generados se utilizaban para evaluar cómo el modelo aprendía a generar nombres de dominio similares a los dominios normales a lo largo del proceso de entrenamiento. Esta evaluación cualitativa permitía analizar el rendimiento del modelo y observar cómo mejoraba la coherencia y realismo de los dominios generados con el aumento de las épocas de entrenamiento.

CódigofuenteutilizadoparalaCreaciónDelModelo

```
model = keras.Sequential([
    keras.Input(shape=(maxlen, len(chars))),
    layers.Conv1D(128, kernel_size=3, activation='relu'),
    layers.MaxPooling1D(pool_size=2),
    layers.LSTM(128),
    layers.Dense(len(chars), activation='softmax')
])

optimizer = Adam(learning_rate=0.01)
model.compile(loss='categorical_crossentropy', optimizer=optimizer)
```

Figura1. Código fuente del modelo utilizado para la creación del modelo.

Es relevante resaltar que, aunque el modelo recibe una semilla inicial para la generación de dominios, el proceso se basa en la combinación de caracteres para producir nombres de dominio. En consecuencia, los primeros dominios generados tendrán una fuerte relación con la semilla, pero a medida que el proceso avanza, los dominios subsiguientes se verán influenciados por los dominios generados previamente. Esta dinámica crea una relación continua y progresiva entre los dominios generados, lo que resulta en una mayor variedad y complejidad en la secuencia de caracteres utilizada.

Con el código y la arquitectura de la red neuronal mostrada anteriormente, se obtiene un modelo optimizado capaz de generar nombres de dominio similares a los dominios normales. Este modelo cumple con el objetivo del estudio, que es generar dominios que sean difíciles de distinguir de los dominios legítimos, lo que permite explorar y comprender mejor las posibles vulnerabilidades y riesgos asociados a la seguridad informática. La combinación de técnicas como las capas de redes neuronales convolucionales y LSTM, junto con la codificación one-hot y la generación iterativa de dominios, proporciona una potente herramienta para generar nombres de dominio realistas y coherentes, lo que contribuye significativamente al avance en el campo de la seguridad cibernética.

2.5 Pruebas de Detección a los Dominios Generados

Una vez que el modelo fue entrenado y los parámetros ajustados, se llevó a cabo la generación de dominios utilizando una semilla específica. Estos dominios generados fueron posteriormente evaluados mediante el uso de diversos detectores. Para llevar a cabo esta evaluación, se seleccionó un dominio normal de la base de datos original y se generaron un total de 4000 dominios utilizando dicha semilla. A continuación, se presentan 25 ejemplos representativos de los 4000 dominios generados:

Porcióndecódigofuenteutilizadoenelgeneradordedominios.

```
epochs = 20
batch_size = 128

for epoch in range(epochs):
    model.fit(x, y, batch_size=batch_size, epochs=1)
    print()
    print("Generating text after epoch: %d" % epoch)

    start_index = random.randint(0, len(text) - maxlen - 1)
    for diversity in [0.5]:
        print("...Diversity:",
        generated = ""
        sentence = text[start_index : start_index + maxlen]
        print('...Generating with seed: "' + sentence + '"')

        for i in range(400):
            x_pred = np.zeros((1, maxlen, len(chars)))
            for t, char in enumerate(sentence):
                x_pred[0, t, char_indices[char]] = 1.0
            preds = model.predict(x_pred, verbose=0)[0]
            next_index = sample(preds, diversity)
            next_char = indices_char[next_index]
            sentence = sentence[1:] + next_char
            generated += next_char
        print("...Generated: ", generated)
        print()
```

Figura2. Porción de código fuente utilizado en el generador de dominios.

▪ aatuaeic.eo	▪ aa.cx	▪ cbroeis.ae	▪ bask.lo	▪ bgs.-o
▪ aii..uo	▪ bena.io	▪ abrrunec..omg	▪ a3ac..oz	▪ aepan.ec
▪ bstrcuc.oz	▪ aaam..no.u	▪ baekeranr.uccc	▪ 3a-ic.ao	▪ ccressn..oo.r
▪ aatkrst.to	▪ btiareraci..oi	▪ cec..ov	▪ baadpibra..oog	▪ ain.so
▪ aaiaea.cem	▪ oloraoire.com	▪ wbece.com	▪ aaitt.ge	▪ bsleca.eo

Estos ejemplos muestran la diversidad y variedad de los dominios generados a partir de la semilla proporcionada, demostrando la capacidad del modelo para generar nombres de dominio de forma automática y aleatoria.

Una vez generados los dominios, se procedió a evaluar su detección mediante el uso de un Detector de Dominios Generados Algorítmicamente (AGD). Con el objetivo de medir la calidad de las predicciones del detector, se empleó la matriz de confusión y diversas métricas de evaluación. Para llevar a cabo esta evaluación, se utilizó el modelo predictor propuesto por Catania et al. [7]. Cabe resaltar que este detector ha demostrado resultados prometedores en investigaciones previas, lo que lo hace una elección adecuada para evaluar la efectividad del modelo de generación de dominios en cuestión.

La matriz de confusión se muestra en la figura 3, y refleja que, de los 4000 dominios generados, solo 399 fueron correctamente detectados como AGD. Las métricas de evaluación obtenidas se muestran en la tabla 2.

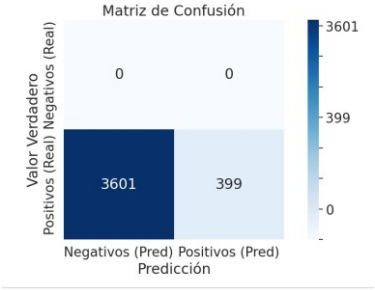


Figura3. Matriz de Confusión

Tabla 2. Métricas de evaluación del Detector de Dominios Generados Algorítmicamente (AGD)

Métrica	Valor
Accuracy (exactitud)	0.10
Precision	1.00
Recall	0.10
F1-score	0.18
Soporte (cantidad de instancias)	4000

Estas métricas proporcionan una evaluación detallada del desempeño del detector de AGD utilizado. La precisión indica la proporción de dominios detectados como AGD que fueron clasificados correctamente. El recall representa la proporción de dominios reales que fueron detectados correctamente como AGD. El F1-score combina la precisión y el recall en una única métrica que pondera ambas medidas. El soporte indica la cantidad total de instancias evaluadas.

Resumiendo, si vemos estos resultados en conjunto, el detector tuvo un 100% de precisión, lo que significa que cuando dijo que algo era un dominio generado algorítmicamente, generalmente tenía razón. Pero, desafortunadamente, solo logró encontrar el 10% de los dominios que realmente eran generados de esta forma. Esto muestra que el detector necesita mejorar para ser más eficaz en la detección de este tipo de dominios.

2.6 Análisis Estadístico de los Dominios

El estudio de las características estadísticas de los dominios proporciona una visión esencial sobre las tendencias en los nombres de dominio. Al analizar la frecuencia de aparición de caracteres, se identificaron patrones significativos. El carácter más recurrente es ’o’, presente en un total de 6.265 ocasiones, seguido de cerca por el punto ’.’ con 6.262 apariciones. Asimismo, la letra ’c’ se presenta comúnmente con 5.596

apariciones, seguida de la letra 'a' con 4.746 ocurrencias. Por otro lado, se encontraron letras menos frecuentes, como la 'x', que solo aparece 5 veces, y el guión bajo ' ' con tan solo 1 aparición.

Otra característica relevante es la longitud promedio de los dominios, la cual se calculó en aproximadamente 10,51 caracteres. Este dato es valioso, ya que nos brinda una comprensión general de la extensión típica de los nombres de dominio estudiados. Un promedio de 10,51 caracteres sugiere que los nombres de dominio tienden a ser relativamente cortos, lo que facilita su memorización y escritura.

El análisis de la frecuencia de aparición de n-gramas (n = 2) también aporta información valiosa sobre la presencia y repetición de ciertas combinaciones de caracteres en los dominios analizados. Algunos de los n-gramas más frecuentes incluyen c¸cc¸on 1.746 apariciones, "..ƨ ".o¸con

1.377 y 1.391 apariciones, respectivamente. Asimismo, se encontraron combinaciones como .ºo¸con 1.124 apariciones, .ªa¸con 769 apariciones y c¸o¸con 765 apariciones. Estos n-gramas pueden revelar patrones comunes en los nombres de dominio, lo que podría facilitar el análisis y la clasificación de futuros dominios.

Por último, se examinó la distribución de caracteres en las posiciones iniciales y finales de los dominios. En la posición inicial, los caracteres más frecuentes son 'c', 'a' y 'b', mientras que los menos comunes son 'j', '6', 'y' y 'z'. En cuanto a la posición final, el carácter más recurrente es 'o', seguido por 'm' y 't'. Por otro lado, los caracteres 'w', '9', '3' y 'x' presentan la menor frecuencia al final de los dominios.

En conjunto, este análisis estadístico nos proporciona una visión más clara de las características de los nombres de dominio examinados, permitiéndonos comprender mejor las tendencias y particularidades que influyen en la elección de estos identificadores en la web.

3. Conclusiones

En conclusión, este estudio investigó la generación de dominios utilizando una red neuronal híbrida compuesta por capas de convolución neuronal (CNN) y memoria de corto-largo plazo (LSTM). Los resultados obtenidos demostraron que los dominios generados presentaron una notable similitud con los dominios legítimos, lo que dificultó su detección por parte del detector de Dominios Generados Algorítmicamente (AGD) utilizado en las pruebas.

El enfoque de generación de dominios utilizado en este estudio tiene el potencial de identificar vulnerabilidades en los sistemas de detección de AGD y mejorar las técnicas de defensa contra ataques de botmasters. La combinación de capas CNN y LSTM en la red neuronal proporcionó resultados prometedores, lo que sugiere que este enfoque puede ser explorado aún más en futuras investigaciones.

Además, los dominios generados en este estudio sirven para aumentar la base de datos de dominios maliciosos, lo cual puede ser de gran utilidad en la preparación y desarrollo de detectores más robustos en investigaciones futuras.

La estadística muestra la frecuencia de apariciones de diferentes dominios de nivel superior (TLDs) en un conjunto de datos. El dominio ".com.[en]cabeza la lista con 506

apariciones. Luego, los dominios .ºoˀ .ᵉᵒ” tienen 255 y 212 apariciones, respectivamente. Se puede observar que la mayoría de los TLDs tienen menos de 100 apariciones, lo que sugiere que hay una amplia variedad de TLDs utilizados en este conjunto de datos. Algunos TLDs menos comunes tienen solo una aparición. Es importante destacar que hay TLDs conocidos como ”.orgˀ”.net¸con solo 6 y 1 apariciones, respectivamente.

En resumen, este trabajo contribuye al campo de la detección de AGD al presentar un enfoque para la generación de dominios que no son fácilmente detectables. Se espera que estos hallazgos impulsen el desarrollo de nuevas estrategias de defensa y fortalezcan la seguridad informática en la lucha contra los ataques de botmasters. Se reconoce que aún existen desafíos en la detección de dominios generados algorítmicamente, y se alienta a futuras investigaciones a seguir avanzando en este campo para asegurar la efectiva protección de los sistemas informáticos.

Referencias

1. Plohmann, D., Yakdan, K., Klatt, M., Bader, J., Gerhards-Padilla, E.: A comprehensive measurement study of domain generating malware. In: 25th USENIX Security Symposium (USENIX Security 16), pp. 263–278. USENIX Association, Austin, TX (2016)
2. Wang, Z., Guo Y.: Neural networks based domain name generation. Journal of Information Security and Applications, vol. 61, pp. 102948 (2021)
3. Schüppen S, Teubert D, Herrmann P, Meyer U. FANCI: Feature-based automated nxdomain classification and intelligence. In: Proceedings of the 27th USENIX Security Symposium, pp. 1165–1181. USENIX Association (2018)
4. Vij P, Nikam S, Bhatia A. Detection of algorithmically generated domain names using LSTM. In: 2020 International Conference on Communication Systems and Networks, COMSNETS 2020, pp. 1–6 (2020)
5. Zhou S, Lin L, Yuan J, Wang F, Ling Z, Cui J. CNN-Based DGA detection with high coverage. In: 2019 IEEE International Conference on Intelligence and Security Informatics, ISI 2019, pp. 62–67 (2019)
6. Graves A, Schmidhuber J. Framewise phoneme classification with bidirectional LSTM networks. Proc Int Joint Conf Neural Netw 2005;4(5–6):2047–2052
7. Catania, C., Garcia, S., Torres, P.: An analysis of convolutional neural networks for detecting DGA. In: Proceedings of XXIV Congreso Argentino de Ciencias de la Computación, La Plata, pp. 1–10 (2018)
8. Chollet, F.: Generate text from Nietzsche’s writings with a character-level LSTM. Keras Examples (2015). Disponible en: https://keras.io/examples/generative/lstm_character_level_text_generation/ (Fecha de acceso: 23/06/2023).
9. Catania, C.A.: Using CNN for a Domain name Generation Algorithm (1) (2023). Disponible en: https://harpomaxx.github.io/post/dga-convnet/ (Fecha de acceso: 23/06/2023).
10. Catania, C.A.: Using CNN for a Domain name Generation Algorithm (2) (2023). Disponible en: https://https://harpomaxx.github.io/post/dga-convnet-p2/ (Fecha de acceso: 23/06/2023).
11. Chollet, F.: Deep Learning con Python. En: Aprende Machine Learning con Python, pp. 325–360. Anaya Multimedia (2021). ISBN: 978-84-415-4351-3.
12. Klaus Greff, Rupesh K. Srivastava, and Jürgen Schmidhuber. ”LSTM: A Search Space Odyssey.¨IEEE Transactions on Neural Networks and Learning Systems 28, no. 10 (2017): 2222-2232.

13. Banjori's DGA. 2021, johannesbader.ch/blog/the-dga-of-banjori/ (Accessed April 12, 2021).
14. Conficker's DGA. 2021, //www.honeynet.org/files/KYE-Conficker.pdf (Accessed April 12, 2021).
15. Symantec. (2015). Timba – Spanish speaking banker targets Chilean banks. Symantec Blogs. https://symantec-enterprise-blogs.security.com/blogs/threat-intelligence/timba-spanish-speaking-banker-targets-chilean-banks.
16. Symantec. (2013). CrytoLocker Ransomware. Symantec Blogs. https://symantec-enterprise-blogs.security.com/blogs/threat-intelligence/crypto-ransomware-crime-unstoppable.
17. Ramdo's DGA. 2021, www.securityweek.com/ramdo-click-fraud-malware- continues-evolve (Accessed April 12, 2021).
18. Leyva, Reynier; Catania, Carlos A. (2023). Generador de Dominios. GitHub. URL: https://github.com/reypapin/Generador-de-Dominios (Consultado el 2 de agosto de 2023).
19. Carlos A. Catania, "DGA Detection Dataset," Hugging Face Datasets, 2023. URL: https://huggingface.co/datasets/harpomaxx/dga-detection
20. Diederik P. Kingma and Jimmy Lei Ba. Adam: A Method for Stochastic Optimization."Published as a conference paper at ICLR 2015.
21. TensorFlow.tf.keras.losses.CategoricalCrossentropy. https://www.tensorflow.org/api_docs/python/tf/keras/losses/CategoricalCrossentropy. Accessed August 3, 2023.
22. Aeva, A. (2017). DGA. GitHub. https://github.com/andrewaeva/DGA.
23. Bambenek Consulting. (n.d.). DGA Domain Feeds. https://faf.bambenekconsulting.com/feeds/.
24. Website ranking. (Accessed April 12, 2021.), s3.amazonaws.com/alexa-static/top-1m.csv.zip,

Coordinadores:
- Fernando Bordignon (UNLu)
- Verónica Artola (UNLP)
- Claudia Russo (UNNOBA)

CACiC 2023

CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

XXII Workshop de Tecnología Informática Aplicada en Educación

> Full Papers

WTIAE

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Laboratorio virtual de electrónica básica para alumnos universitarios dentro de aula extendida

Mg. Leonardo Navarria[1,2], Mg. Alejandro González[3,4], Dra. Alejandra Zangara[3]

[1] Facultad de Ciencias Astronómicas y Geofísicas (UNLP)
Paseo del Bosque S/N La Plata
[2] Facultad de Ingeniería (UNLP) 1 esq. 47 La Plata
[3] Instituto de Investigación en Informática III- LIDI
Facultad de Informática (UNLP) 50 esq. 120 La Plata
[4] Dirección General de Educación a Distancia y Tecnologías. UNLP
Argentina Calle 7 nro.776 La Plata
navarrial@fcaglp.unlp.edu.ar

Resumen. El objetivo de este trabajo fue desarrollar laboratorios virtuales de electrónica básica para alumnos universitarios dentro de un aula extendida y evaluar el contenido de estos utilizando el juicio por jurado de expertos. En la enseñanza presencial de carreras que poseen contenido práctico se hace fundamental la realización de prácticas para afianzar conocimientos. La ejercitación en las mismas involucra la asistencia del estudiante a determinados turnos de horarios fijos, en un lugar limitado en recursos de equipamiento y espacio. Con el uso de metodologías de trabajo virtuales se puede lograr que un gran número de alumnos tenga acceso a los contenidos prácticos para poder familiarizarse con instrumentos y componentes de electrónica básica que son de fundamental importancia para la adquisición de conocimientos prácticos. Al final del trabajo se muestran los resultados obtenidos de forma global por la metodología enunciada.

Palabras claves: prácticas, laboratorios virtuales, circuitos, simulación

1. Introducción

El presente trabajo se deriva del trabajo de Tesis la obtención del grado de Magister en Tecnología Informática Aplicada a la Educación dictada en la Facultad de Informática de la Universidad Nacional de La Plata. [1]

El autor es docente de la Facultad de Ingeniería y de la Facultad de Ciencias Astronómicas y Geofísicas, ambas de la UNLP. En las asignaturas que el autor ejerce su cargo docente se dictan contenidos básicos de electrotecnia, como ser leyes fundamentales, comportamiento de componentes eléctricos y electrónicos, análisis de distintos tipos de circuitos como ser en corriente continua como alterna. Por tratarse de materias de contenidos teóricos y prácticos, los análisis de los distintos tipos de circuitos pueden ser verificados utilizando simuladores de circuitos y también implementados de forma práctica, siendo aquí de vital importancia conocer el manejo de los instrumentos que permiten medir variables eléctricas.

En la enseñanza presencial de carreras que poseen contenido práctico se hace fundamental la realización de prácticas para afianzar conocimientos. La ejercitación en las mismas involucra la asistencia del estudiante a determinados turnos de horarios

fijos, en un lugar limitado en recursos de equipamiento y espacio. Con el uso de metodologías de trabajo virtuales se puede lograr que un gran número de alumnos tenga acceso a los contenidos prácticos para poder familiarizarse con instrumentos y componentes de electrónica básica y avanzada.

Para evaluar que el contenido de los laboratorios mejore las competencias de los estudiantes, se validaron los contenidos de cada uno de los laboratorios mediante el método de juicio por jurado de expertos. Con ello se buscó indagar la posibilidad de mejorar los contenidos, verificar si las prácticas de laboratorios mejoraron los procesos cognitivos, emplear simuladores en el proceso de enseñanza y de aprendizaje y evaluar material informativo.

2. Propósito

a. Práctica de Laboratorio

La práctica de laboratorio es una potente estrategia pedagógica para la construcción de competencias procedimentales y por este motivo es utilizada en una gran variedad de programas académicos, usualmente sincronizada con su asignatura teórica correspondiente. El trabajo de Infante Jiménez [2] aborda la importancia los laboratorios presenciales en las carreras de ciencias aplicadas por ejemplo Ingeniería, Fisicoquímica, Química, etc., sin embargo, a medida que los modelos educativos se han vuelto más flexibles y enfocados a competencias, la inclusión de las Tecnologías de la Información y la Comunicación (TIC) ha cambiado radicalmente el concepto de espacio físico; esto ha hecho patente una serie de limitaciones pues a pesar de la enorme importancia que éste tiene para el aprendizaje, no puede ofrecer la versatilidad idónea que se necesita en la actualidad. También es un hecho que la práctica de laboratorio presenta elevados tiempos de respuesta, los cuales tienden a bajar su productividad. Infante Jiménez remarca el costo de los experimentos con plantas reales respecto tiempo, dinero, energía, mantenimiento, condiciones que son difíciles de sostener si no se dispone de personal idóneo con alta disponibilidad frente a la demanda de las prácticas. Junto a esto último se debe tener la supervisión por parte de docentes tutores o encargados de laboratorios. Todo esto conlleva a una limitación física de la cantidad de alumnos. Pensando en una época de pandemia COVID-19 se le tendría que sumar el costo de la sanitización y que los laboratorios deberían ser de sólo un estudiante ya que por protocolo se imposibilita compartir elementos e instrumentos de medición. Sumado a esto mismo el impacto negativo de la huella de carbono ya que el traslado del alumno y docente, salvo que no se utilicen vehículos de combustión, generan emisiones de gases de efecto invernadero a la atmósfera.

El artículo de Contreras Gélvez [3] explora los hábitos relacionados con el uso de simuladores para el apoyo en los procesos de transferencia del conocimiento en ciencias básicas y programación. En este trabajo se parte de la premisa para investigar el aporte de los simuladores a la formación de estudiantes universitarios de ingeniería. El estudio se contextualiza en el desarrollo de clases de los saberes de física, matemáticas y programación. Se contaron con laboratorios físicos dotados de puestos de trabajos suficientes para la cantidad de estudiantes.

La utilización de herramientas de laboratorios virtuales se considera que generará una interacción alumno tutor a través de un entorno virtual formal de enseñanza. El laboratorio virtual como objeto de aprendizaje posee como atributos:

- Ser reutilizables: pueden volver a usarse sin perder ninguna característica.
- Ser digital.
- Ser un recurso: no constituye en sí mismo el conocimiento, sino que lo promueve y lo fortalece de manera más fácil.
- Servir de aprendizaje: hace hincapié a la característica de explícito e intencionado de que la persona debe aprender a través del objeto de aprendizaje

Como resultado "se puede decir, en forma general, que el uso de simuladores como estrategia didáctica, a través de los cuales se transfiere conocimiento, sí causa impacto en el proceso de aprendizaje de los estudiantes, ya que las clases se vuelven más interesantes, existe una mayor participación de los alumnos, son más claras las explicaciones que se dan, incrementan la retención al presentarse los contenidos, y aumenta la motivación y el gusto por aprender" [2]. Se debe tener en cuenta que la utilización de simulaciones como herramienta de enseñanza presenta múltiples ventajas, sin embargo, su integración en diseños didácticos concretos requiere que los profesores conozcan las características, las potencialidades y las limitaciones de estos recursos.

En la enseñanza de la ingeniería y carreras técnicas, es deseable la realización de actividades pedagógicas que le permitan al estudiante poner en práctica los conocimientos previos y adquirir nuevos, y corroborados en el campo de la experiencia real. Este papel lo desempeña el laboratorio de prácticas, el cual, inexorablemente, requiere de la presencia física del estudiante para poder manipular los sistemas de control y las plantas existentes en un entorno controlado, bajo la supervisión del profesor [4].

b. Aprendizaje Basado en Problemas

El Aprendizaje Basado en Problemas (ABP) como metodología de enseñanza y de aprendizaje es utilizado en numerosas instituciones de educación superior en diversas áreas del conocimiento. La finalidad del Aprendizaje Basado en Problemas es formar estudiantes capaces de analizar y enfrentarse a los problemas de la misma manera en que lo hará durante su actividad profesional, es decir, valorando e integrando el saber que los conducirá a la adquisición de competencias profesionales.

A medida que avancen en la carrera utilizar el ABP como método de enseñanza aprendizaje centrado en el estudiante, le permitirá la adquisición de conocimientos, habilidades y actitudes a través de situaciones de la vida real. La no enseñanza de la aplicación del ABP hará que el estudiante no pueda construir su conocimiento sobre la base de problemas y situaciones de la vida real y que, además, no lo haga con el mismo proceso de razonamiento que utilizará cuando sea profesional.

La característica más innovadora del ABP es el uso de problemas como punto de partida para la adquisición de conocimientos nuevos y la concepción del estudiante como protagonista de la gestión de su aprendizaje

La mejora de la resolución de situaciones utilizando ABP podrá favorecer en cuestiones fundamentales como ser:

- Unificación de criterios respecto a la resolución de problemas
- Unificación de lenguaje
- Resolución de situaciones
- Fundamentación correcta
- Adquisición de criterios para ejercer como futuro profesional.

En las asignaturas de trabajo la metodología ABP se utiliza para resolver problemas de trabajos prácticos orientados a situaciones reales que el estudiante puede encontrar en su vida profesional.

c. Aprendizaje fundamentado en trabajos de laboratorio

De acuerdo con López Rúa y Tamayo [6] las simulaciones y los trabajos prácticos de laboratorio o trabajos experimentales constituyen una de las actividades más importantes en la enseñanza de las ciencias porque promueven la adquisición de una serie de procedimientos y habilidades científicas, desde las más básicas (utilización de multímetros, medición de variables, toma de datos, tratamiento de datos, etc.) hasta las más complejas (investigar y resolver problemas haciendo uso de la experimentación). Esto indica la fundamental importancia que los trabajos prácticos deben tener como actividad de aprendizaje.

En el área de las ciencias el trabajo de laboratorio favorece y promueve el aprendizaje, dado que el estudiante confronta sus saberes previos con la realidad al verificarlos mediante las prácticas [7].

La actividad experimental no solo debe ser vista como una herramienta de conocimiento, sino como un instrumento que promueve los objetivos conceptuales, procedimentales y actitudinales que debe incluir cualquier propuesta pedagógica. Siguiendo esa línea de pensamiento, Andrés, Pesa y Meneses [8] proponen que en el desarrollo de un trabajo práctico de laboratorio debería predominar el aprendizaje en el dominio metodológico y en una inseparable relación con algún marco teórico asociado a la experiencia planteada, el estudiante aprende en acción frente a la situación.

La permanente investigación, innovación, integración y desarrollo de los sistemas tecnológicos e informáticos en el ámbito de la educación están permitiendo la implementación de nuevas prácticas en la forma de realizar la construcción del proceso de enseñanza y aprendizaje. Tal es el caso de la realización de un trabajo práctico de laboratorio ya sea sobre un objeto de existencia virtual o real pero efectuada en forma remota, como una actividad a la que se accede a través de un entorno virtual de enseñanza aprendizaje.

Se puede afirmar que los trabajos de laboratorio, dependiendo del grado de participación del estudiante establece que:

Se involucra al estudiante en el uso de procedimientos técnicos y científicos

- Se establece una metodología de trabajo
- Se requiere el uso de material e instrumentos específicos, y en algunos casos reacondicionados y adaptados para que el estudiante los utilice para llevar adelante

las prácticas.

- Se realizan en lugares distintos al aula dónde se dictan las clases, pudiendo ser estos un laboratorio, situando en el caso de estudio, de electrónica o directamente campo como es el caso de geofísica, laboratorio para medir el campo magnético terrestre.
- Se presentan riesgos para el alumno respecto de la manipulación de algunos componentes.
- Se presentan riesgos de daños de instrumentos de medición
- Se realizan actividades que poseen una complejidad de organización superior a la resolución de un ejercicio en el material de estudio.
- Se genera una motivación por parte del estudiante.
- Se ayuda a la comprensión de los contenidos teóricos expuestos en el aula.
- Se fomenta el desarrollo del razonamiento científico.
- Se facilita la comprensión de problemas reales modelizados para la resolución en carpeta de estudio.
- Se establece una base de contenidos sobre la que se desarrollan actitudes de investigación.
- Deben ser fundamentales para la enseñanza y el aprendizaje de conocimientos técnicos.

3. Desarrollo

El fenómeno de los avances tecnológicos digitales en el campo de la comunicación y de la informática, ha enriquecido los instrumentos de la tecnología educativa con la incorporación de los equipos de grabación, reproducción y transmisión de texto, sonido e imagen. Y no olvidemos que buena parte de estas tecnologías se encuentran hoy en los hogares de multitud de ciudadanos.

Así, los alumnos, a través del estudio independiente, aprenderán al menos de forma tan eficaz como el estudiante de un centro convencional. Más que el método en sí importa el buen uso que se haga del mismo como muestran numerosas investigaciones. La educación en la virtualidad, es decir, desde la no-presencia en entornos virtuales de aprendizaje, no se sitúa necesariamente en ninguna orientación educativa concreta. Al igual que en la presencialidad existe la convivencia entre orientaciones y didácticas diversas, siempre que éstas actúen de forma coherente con las finalidades educativas y con los fines de la educación, de la misma forma sucede en la virtualidad. [9]. El proceso de aprendizaje en ambientes virtuales es el resultado de varias etapas en las que el alumno construye su aprendizaje. Asimismo, puede ser el producto de la práctica, como puede ser el empleo de simuladores. De esta forma existe un análisis crítico como también lo existe en la presencialidad dando lugar a diferentes perspectivas de valoración.

La diferencia de mayor importancia entre una educación presencial y virtual es el cambio de medio y el tratamiento que se debe hacer para optimizar cada uno de los medios. No se debe ni se puede realizar las mismas acciones en medios distintos,

aunque nuestros objetivos educativos sean los mismos. Se debe saber que los caminos a recorrer son distintos. Es de fundamental importancia el medio de comunicación, pudiendo llevar al éxito o al fracaso del proceso educativo.

A continuación, se lista alguno de los laboratorios realizados junto con sus enlaces:
Análisis de circuitos en corriente continua, figura 1:
https://lc.cx/ZvBX8x

Para este laboratorio se utilizó para el diseño web la herramienta de genial.ly. Se utilizaron componentes físicos de circuitos eléctricos como protoboard, resistores, cables, etc. Se realizaron varias mediciones con instrumentos de medida de variables eléctricas para que el estudiante vincule el circuito real con un esquemático.

Los objetivos fueron: Verificar experimentalmente las leyes de Kirchhoff de las corrientes y las tensiones. Aprender a realizar e interpretar medidas de corrientes y tensiones en Corriente Continua. Comprender los conceptos de circuitos eléctricos y potencia. Vincular un circuito real con un esquemático. Realizar cálculos y comprobar los resultados obtenidos

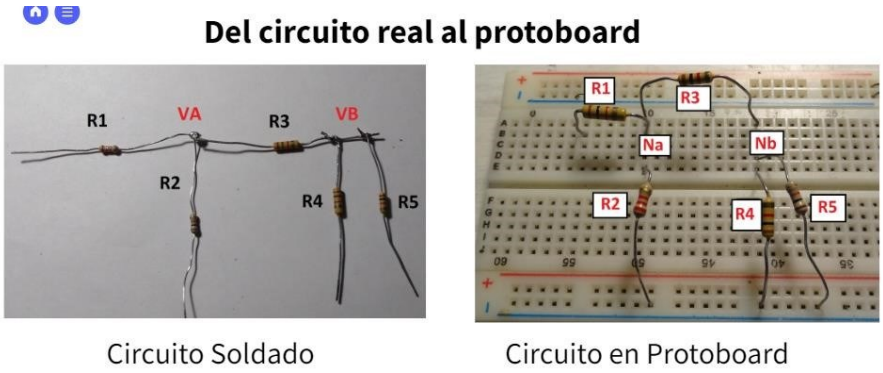


Figura 1: Circuito soldado y circuito equivalente montado en un protoboard

Análisis de circuitos en corriente alterna:
https://youtu.be/Z6-fXSLZ1Pc

Se utilizaron componentes físicos de circuitos eléctricos, resistores, cables, inductores y capacitores. Se utilizó un multímetro, osciloscopio (figura 2) y un generador de señales.

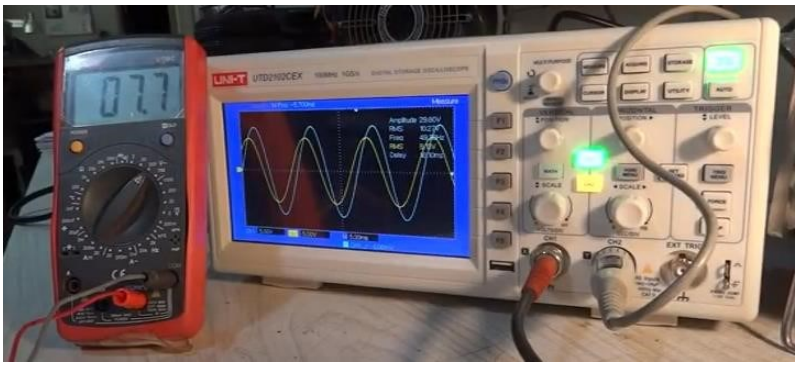


Figura 2: Osciloscopio y multímetro para la medición de corriente alterna

Los objetivos fueron: a. Interpretar el uso de generadores de señales. Comprender el uso del multímetro. Configurar el uso de un osciloscopio

Laboratorios de Rectificadores, figura 3:
https://youtu.be/k3vcU75SsUw
https://youtu.be/0KVOVrWAToU
https://youtu.be/SAEMP_zOwng

Este laboratorio tuvo como objetivos: Autoevaluar los conocimientos del alumno sobre el tema rectificadores. Comprender el funcionamiento de los rectificadores estudiando en tiempo real con mediciones de instrumentos de medida. Comprobar los resultados obtenidos utilizando el simulador de circuitos eléctricos LTSPICE. Obtener conclusiones de los resultados obtenidos.

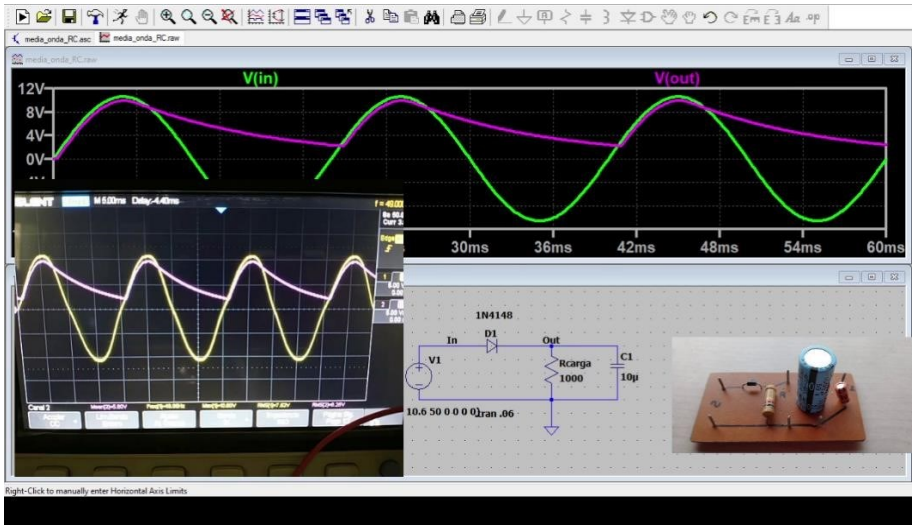


Figura 3: Laboratorio de Rectificación

Para este tercer laboratorio se utilizó un Branching scenario (H5P). Esta herramienta permite tomar decisiones sobre el comportamiento de un determinado circuito. Asimismo, se utilizó un simulador de circuitos LTspice, junto al circuito real, instrumentos de medición y se filmó la realización de los laboratorios, comparando los resultados obtenidos con los simulados como muestra la figura 3.

4. Evaluación de los contenidos de los laboratorios

El objetivo general del trabajo que desarrolla el tesista fue evaluar la efectividad de realizar prácticas de electrónica básica con laboratorios virtuales dentro de un aula extendida. La evaluación se realizó por jurado de expertos, que consiste en solicitar a una serie de personas la demanda de un juicio hacia un objeto, un instrumento, un material de enseñanza, o su opinión respecto a un aspecto concreto. Cada uno de los laboratorios realizados fue pensado para que el alumno mejore las competencias para la resolución de problemas de cada una de las asignaturas en cuestión. . Para seleccionar los expertos acordes a lo indicado en los párrafos anteriores se tomaron en cuenta:

- Experiencia profesional.
- Vinculación a la investigación.
- Experticia en el tema.
- Nivel Académico.
- Participación en proyectos de investigación

Para llevar adelante el juicio por expertos se utilizó la técnica de Agregados Individuales, que consiste en que cada experto responde por escala de Likert una serie de preguntas acerca de las distintas dimensiones de los laboratorios. El método de agregados individuales permite que los expertos sean consultados de forma individual y simultánea mediante un formulario electrónico. Los autores Corral [10] , Cabero [11] coinciden en que la construcción del instrumento de indagación se convierte en una herramienta clave, ya que no habría otra interacción con los expertos más que el envío y recepción del instrumento completo. Los expertos evalúan de forma individual y no

pueden intercambiar sus opiniones, puntos de vista y experiencia con los restantes expertos seleccionados. Para la evaluación de los materiales construidos se atacaron los ítems que se muestran en el siguiente listado:

1. Calidad de contenidos: Actualización, Calidad, Secuencia y estructura, Claridad de explicaciones.
2. Aspectos técnicos: Calidad de medios, Tamaños de gráficos y letras.
3. Motivación: Grado de atracción de la herramienta: Interés que despierta, Duración, Alcance de objetivos
4. Valor didáctico: Adaptación al Currículo, Favorece el proceso de aprendizaje, Adecuación del vocabulario, Explicación de los objetivos, Las problemáticas presentadas corresponden con situaciones reales, Conclusiones.

Se convocaron a 15 jurados de los cuales 13 respondieron el cuestionario vencido el plazo, que fue aproximadamente de unos 30 días. Por lo tanto, el grado de convocatoria arroja un 87% de convocatoria efectiva, perdiendo solamente una muestra

del 13%. En la figura 4 se muestran los resultados globales obtenidos en el Laboratorio comprobación de las Leyes de Kirchhoff por cada uno de los ítems aplicada la escala de Likert. Los resultados se presentan de forma porcentual.

Item	Muy Bueno	Bueno	Suficiente	Insuficiente
Alcance de objetivos	62%	38%	0%	0%
Adaptación al currículo	62%	38%	0%	0%
Favorecimiento del proceso de aprendizaje	69%	23%	8%	0%
Adecuación del vocabulario	54%	38%	8%	0%
Explicación de los objetivos	77%	15%	8%	0%
Correspondencia entre situaciones reales	69%	31%	0%	0%
Conclusiones obtenidas	62%	31%	0%	8%
Actualización de contenidos	54%	31%	15%	0%
Calidad del diseño	46%	46%	8%	0%
Secuencia y estructura	46%	46%	8%	0%
Claridad de explicaciones	69%	31%	0%	0%
Calidad de los medios	38%	62%	0%	0%
Tamaño de gráficos y letras	77%	23%	0%	0%
Grado de atracción de la herramienta	69%	23%	8%	0%
Interés despertado	62%	31%	8%	0%
Duración	69%	23%	8%	0%
Promedio	**62%**	**33%**	**5%**	**0%**

Figura 4 Resultados Obtenidos de la comprobación de las Leyes de Kirchhoff

Para el laboratorio de comprobación de leyes fundamentales de electricidad como son las Leyes de Kirchhoff que el valor de aceptación es del 95% repartidos entre Muy Bueno y Bueno. No obstante, se toman notas de los aportes indicados por algunos de los expertos en el caso de las conclusiones obtenidas, la duración del laboratorio y el interés que despierta este primer laboratorio.

En la figura 5 se muestran los resultados que se obtuvieron del segundo laboratorio, utilización de instrumentos de medición de variables eléctricas en corriente alterna.

Ítem	Muy Bueno	Bueno	Suficiente	Insuficiente
Alcance de objetivos	62%	38%	0%	0%
Adaptación al currículo	62%	38%	0%	0%
Favorecimiento del proceso de aprendizaje	85%	8%	8%	0%
Adecuación del vocabulario	62%	38%	0%	0%
Explicación de los objetivos	62%	38%	0%	0%
Correspondencia entre situaciones reales	62%	31%	8%	0%
Conclusiones obtenidas	69%	23%	8%	0%
Actualización de contenidos	69%	23%	8%	0%
Calidad del diseño	62%	31%	0%	8%
Secuencia y estructura	69%	15%	15%	0%
Claridad de explicaciones	69%	31%	0%	0%
Calidad de los medios	69%	23%	8%	0%
Tamaño de gráficos y letras	69%	23%	8%	0%
Grado de atracción de la herramienta	46%	46%	8%	0%
Interés despertado	62%	23%	8%	8%
Duración	69%	15%	15%	0%
Promedio	**65%**	**28%**	**6%**	**1%**

Figura 5 instrumentos de medición de variables eléctricas en corriente alterna

Para el laboratorio de uso de instrumentos de medición se observa posee una aceptación del 93%. Existen puntos a revisar respecto de las conclusiones obtenidas y la motivación, debiendo esto ajustar la motivación de realizar este laboratorio agregándole un grado de justificación más conceptual.

La figura 6 muestra los resultados que se obtuvieron del tercer laboratorio, rectificadores.

Ítem	Muy Bueno	Bueno	Suficiente	Insuficiente
Alcance de objetivos	92%	8%	0%	0%
Adaptación al currículo	62%	38%	0%	0%
Favorecimiento del proceso de aprendizaje	77%	23%	0%	0%
Adecuación del vocabulario	69%	31%	0%	0%
Explicación de los objetivos	62%	38%	0%	0%
Correspondencia entre situaciones reales	85%	15%	0%	0%
Conclusiones obtenidas	77%	23%	0%	0%
Actualización de contenidos	62%	38%	0%	0%
Calidad del diseño	77%	23%	0%	0%
Secuencia y estructura	62%	38%	0%	0%
Claridad de explicaciones	69%	31%	0%	0%
Calidad de los medios	46%	54%	0%	0%
Tamaño de gráficos y letras	62%	38%	0%	0%
Grado de atracción de la herramienta	77%	23%	0%	0%
Interés despertado	69%	31%	0%	0%
Duración	54%	46%	0%	0%
Promedio	**69%**	**31%**	**0%**	**0%**

Figura 6 Resultados del laboratorio de rectificadores.

Para el laboratorio de Análisis de Rectificadores se obtuvo un 100% de aceptación por parte de los jurados.

5. Conclusiones

A través de los laboratorios se propuso mejorar las competencias en el área del conocimiento tratado. La evaluación de las competencias aporta un cambio de enfoque respecto a la evaluación tradicional, ya que intenta determinar los conocimientos y las aptitudes logradas en el proceso de aprendizaje.

La propuesta pedagógica para la inclusión del laboratorio virtual en un esquema tradicional de cursada contempló la etapa de experiencia real, experiencia virtual, simulación, comparación de resultados y obtención de conclusiones.

El criterio del docente es determinante para la selección del laboratorio virtual que mejor se acople a los objetivos acordes a la formación por competencias. Los diseños de los laboratorios han requerido un esfuerzo extra por parte del docente.

Es necesario que se continúe fomentando estudios que contribuyan al conocimiento sobre el uso de las tecnologías en la educación y en la transferencia de conocimiento.

Referencias

1. Navarria L. (2023) Laboratorio virtual de electrónica básica para alumnos universitarios dentro de aula extendida. Comprobaciones de las leyes básicas de circuitos eléctricos y aplicaciones sobre semiconductores http://sedici.unlp.edu.ar/handle/10915/154689
2. Jiménez, C. I. (2014). Propuesta pedagógica para el uso de laboratorios virtuales como actividad complementaria en las asignaturas teórico-prácticas. *Revista mexicana de investigación educativa*, 917-937. Obtenido de http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1405-66662014000300013&lng=es&tlng=es
3. Contreras Gelves, Gloria Amparo, & García Torres, Rosa, & Ramírez Montoya, María Soledad (2010). Uso de simuladores como recurso digital para la transferencia de conocimiento. Apertura, 2. ISSN: 1665-6180. Disponible en: https://www.redalyc.org/articulo.oa?id=688/68820841008
4. Lorandi, A.P.; Hermida, G.; Hernández, J., y Ladrón de Guevara, E. (2011). Los Laboratorios Virtuales y Laboratorios Remotos en la Enseñanza de la Ingeniería. Revista Internacional de Educación en Ingeniería, 4, 24-30. Recuperado de http://bibliografia.eovirtual.com/LorandiA_2011_Laboratorios.pdf
5. Dormido, S. & Sanchez, Jhonnatan & Morilla, F. (2007). Laboratorios virtuales y remotos para la práctica a distancia de la Automática.
6. López Rua, Ana Milena, & Tamayo Alzate, Óscar Eugenio (2012). Las Prácticas De Laboratorio En La Enseñanza De Las Ciencias Naturales. Revista Latinoamericana de Estudios Educativos (Colombia), 8(1),145-166. ISSN: 1900-9895. Disponible en: https://www.redalyc.org/articulo.oa?id=134129256008.
7. Osorio, Y.W. (2004). "El experimento como indicador de aprendizaje". Boletín PPDQ, No. 43, pp. 7-10.
8. Andrés, M. M., Pesa, M. Y Meneses, J. (2008). Efectividad de un laboratorio guiado por el modelo de aprendizaje matlaf para el desarrollo conceptual asociado a tareas experimentales. Enseñanza de las Ciencias, 2008, 26(3), 343-358
9. Sangrà, Albert. (2001). Enseñar y aprender en la virtualidad. http://www.redined.mec.es/oai/indexg.php?registro=007200230138.
10. Corral, Yadira. Validez y confiabilidad de los instrumentos de investigación para la recolección de datos. Revista Ciencias de la Educación. 19.2009
11. Cabero Almenara, J. y Llorente Cejudo, M. C. (2013), La aplicación del juicio de experto como técnica de evaluación de las tecnologías de la información (TIC). En Eduweb. Revista de Tecnología de Información y Comunicación en Educación, 2012. Disponible en http://tecnologiaedu.us.es/tecnoedu/images/stories/jca107.pdf

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Propuesta STEAM para la formación científica de estudiantes pertenecientes a carreras del área de las ciencias exactas y naturales

Silvina Manganelli[1,2]

[1] ITU, Instituto Tecnológico universitario, Universidad Nacional de Cuyo
[2] FCEN, Facultad de Ciencias Exactas y Naturales, Universidad Nacional de Cuyo
smanganelli@fcen.uncu.edu.ar

Resumen. El modelo de enseñanza STEM (Ciencia, Tecnología, Ingeniería, Matemática) posee como objetivos mejorar el rendimiento de los estudiantes en Ciencias, Tecnología, Ingeniería y Matemáticas y comprender y aplicar de manera integrada la Ciencia, la Tecnología, la Ingeniería y las Matemáticas. La presente propuesta plantea la necesidad de ampliar las prácticas educativas, incluyendo el desarrollo de proyectos tecnológicos interdisciplinares conformes al modelo de enseñanza STEAM. Esta actividad comprende el desarrollo de un proyecto interdisciplinar, en el cual los estudiantes parten de un problema, para resolverlo, primero lo cuestionarán, realizaran indagaciones, consultaran trabajos y papers actuales relacionados a la temática, formularán preguntas. Luego de esas preguntas, obtendrán una idea que se convertirá en hipótesis. Luego llevarán a cabo una experimentación científica con el objeto de probar su hipótesis. Para ello realizarán mediciones e indagaciones, hasta convertir su idea original en un producto que puede ser lógico (programa o aplicación) o físico (dispositivo).

Palabras clave: STEAM, ECBI, ABP, MAKER, PENSAMIENTO COMPUTACIONAL, ARDUINO

1. Introducción

Los objetivos del modelo de enseñanza STEAM (Ciencia, Tecnología, Ingeniería, Arte Matemática) buscan: incrementar la alfabetización STEM (Ciencia, Tecnología, Ingeniería, Matemática) del estudiantado y promover actitudes positivas hacia las disciplinas implicadas [1][2][3]

La competencia científica se concibe como un continuo que abarca desde los niveles de competencia científica más bajos hasta los más avanzados. Dicho de otra manera, se considera que las personas poseen diversos grados de competencia científica y no que posean o carezcan de competencia científica en términos absolutos Por ejemplo, un estudiante con un nivel de competencia menos desarrollado puede ser capaz de recordar conocimientos científicos sencillos y de emplear conocimientos científicos de uso corriente para sacar y evaluar conclusiones. En cambio, un alumno con un nivel de competencia científica más avanzado podrá crear y emplear modelos con el objeto de hacer predicciones y dar explicaciones, analizar investigaciones científicas, relacionar entre sí datos que puedan constituirse en pruebas, evaluar explicaciones alternativas de un mismo fenómeno y exponer sus conclusiones con precisión

Los objetivos STEM presentan una naturaleza política y educativa [1]. Los objetivos políticos apuntan al incremento de la competitividad económica de los países.

Mientras que los enmarcados en el ámbito educativo; comprenden una perspectiva social y otra académica. Los objetivos del modelo de enseñanza STEAM señalan también un ajuste del currículum a las condiciones laborales y a las características de la sociedad actual. También buscan [1]:

- Desarrollar estudiantes capaces de participar críticamente en un mundo altamente tecnológico y globalizado
- Introducir en las escuelas las habilidades y los conocimientos que son cada vez más importantes para las necesidades sociales actuales.
- Despertar interés hacia estas disciplinas.
- Comprender y aplicar de manera integrada la Ciencia, la Tecnología, la Ingeniería y las Matemáticas.
- Mejorar el rendimiento de los estudiantes en Ciencias, Tecnología, Ingeniería y Matemáticas.

Acorde a este modelo STEAM para la enseñanza, la presente propuesta plantea la necesidad de ampliar las prácticas educativas, las cuales deberían incluir el desarrollo de proyectos tecnológicos interdisciplinares conformes a este modelo de enseñanza STEAM.

Este modelo de enseñanza STEAM podría enmarcarse dentro del aprendizaje basado en proyectos (ABP), con la particularidad de que la solución al problema planteado suele ser un objeto tecnológico (un dispositivo, un programa, etc.) [4]. Entre las metodologías activas, las más adecuadas para el desarrollo de las competencias STEAM son: el aprendizaje basado en problemas, el movimiento maker y el pensamiento computacional como estrategias de enseñanza aprendizaje [4][5].

Esta actividad forma parte de una serie de actividades enmarcadas dentro del modelo de enseñanza STEAM, las cuales se investigan y ejecutan dentro del espacio curricular Pensamiento Computacional, Programación y STEAM, correspondiente al ciclo orientado de los profesorados que se imparten en la Facultad de Ciencias Exactas y Naturales de la Universidad Nacional de Cuyo. Cada actividad comprende el desarrollo de un proyecto interdisciplinar, que abarca dos o más espacios curriculares del área de las ciencias, sumados a las áreas propias de STEAM. De esta manera los estudiantes parten de un problema, y para resolverlo, primero lo cuestionarán, realizaran indagaciones, consultaran trabajos y papers actuales relacionados a la temática y formularán preguntas. Luego de esas preguntas, sumado a una fase creativa de ideación, obtendrán una idea que se convertirá en hipótesis. Posteriormente llevarán a cabo una experimentación científica con el objeto de probar su hipótesis. Para ello realizarán mediciones, indagaciones, hasta convertir su idea original en un producto que puede ser lógico (programa o aplicación) o físico (dispositivo).

1.1 Objetivos

Esta propuesta tiene como objetivos:

- Estimular el desarrollo de competencias científicas, digitales, y las necesarias para abordar los desafíos del siglo XXI.
- Desarrollar en los estudiantes habilidades transversales como: la investigación, el

pensamiento crítico, la resolución de problemas, la creatividad, la comunicación y la colaboración.

- Afianzar los conceptos y contenidos de los espacios curriculares involucrados en la actividad para llevar a cabo la experiencia científica propuesta.

2. Desarrollo

La enseñanza de las ciencias basada en la indagación (ECBI) propone que el estudiantado aprenda ciencias emulando el proceso investigativo que sigue la ciencia para crear conocimiento: formular una pregunta investigable, diseñar experimentos y/o recoger y analizar datos, sacar conclusiones de datos, formular una explicación [4].

La actividad STEAM propuesta en el presente trabajo incorpora el uso de una TIC muy emblemática dentro del mundo maker, la plataforma de hardware de código abierto, Arduino.

A continuación se describe la actividad STEAM propuesta basada en la indagación y la experimentación científica.

2.1 Actividad STEAM propuesta: Relación SUELO - CLIMA – PLANTA

Se realizó un proyecto en el cual los estudiantes debían indagar y poner en manifiesto los conocimientos o saberes aprendidos en espacios curriculares como: Biología Vegetal, Informática, Física I, Ciencias de la Tierra, Introducción al pensamiento computacional, programación y actividades STEAM.

A continuación se plantea un problema que coloca al riego de una planta como protagonista de una experiencia STEAM.

La actividad parte de un problema del entorno social. Para resolverlo el estudiante desarrollará una solución tecnológica a partir de una experiencia científica que comprende el manejo de conocimientos relacionados con el suelo, clima y planta y el uso de recursos tecnológicos populares como las placas electrónicas Arduino.

La planta o el cultivo constituye una unidad o elemento conductor de agua entre el suelo y la atmósfera, absorbiéndola desde el suelo, para luego circular por el xilema o capilar y perdiéndose a través de los estomas de las hojas, hacia la atmósfera por un proceso conocido como evapotranspiración [6]

Problema:

La situación hídrica en Mendoza es crítica desde hace años. Las características climatológicas de la región, las pocas lluvias y nevadas en la temporada invernal y el bajo nivel de sus reservas para abastecer a la población cuando comienza el verano son una constante en la provincia. A ese marco poco alentador se le suma la falta de conciencia colectiva por el uso de este recurso tan valioso.

Respecto al riego de las plantas, ¿de qué manera piensas podrías colaborar con esta problemática? ¿Que aporte podría sumar a la conciencia colectiva?

2.2 Objetivos de la actividad:

- Resolver situaciones problemáticas que demanden el estudio de variables pertenecientes al suelo, clima, planta, a través de una solución tecnológica que optimice el proceso de recolección de datos o el proceso de modelado de los datos obtenidos.
- Integrar a la actividad propuesta; herramientas que faciliten el acceso a los datos experimentales, enriquezcan su análisis y favorezcan los tiempos de recolección de datos.
- Mejorar los aprendizajes y rendimiento de los estudiantes en Ciencias, Tecnología, Ingeniería y Matemáticas

En este proyecto integrado intervienen saberes que los estudiantes pueden indagar en los espacios curriculares que han transitado a lo largo de su carrera: Biología Vegetal, Ciencias de la Tierra, Física I, Informática, Introducción al pensamiento computacional, programación y actividades STEAM.

2.3 Exploración del problema. Inmersión.

Se plantea a los estudiantes la siguiente propuesta ¿Qué aporte o solución puedes brindar desde la apropiación social de los conocimientos que obtuviste en los espacios curriculares de ciclo básico física, biología, química, matemática, geología, para colaborar con un uso adecuado del agua?

Fase creativa "la ideación"

En este momento surgen de la imaginación de los estudiantes muchas preguntas y muchas ideas. Se pretende que estas preguntas e ideas no estén filtradas y que el proceso creativo se estimule con alguna dinámica previa que abra espacio a la imaginación, como la técnica del pensamiento lateral.

Preguntas

La exploración de este tema permite a los estudiantes indagar nuevos aspectos o perspectivas sobre el mismo. El estudiante consultará papers, que lo ayuden a sostener su futura hipótesis. Esta indagación es una consecuencia natural de nuestra innata curiosidad y deseo de descubrir. Estas preguntas impulsaron o activaron el deseo de explorar, descubrir e investigar.

Para registrar, publicar y compartir las preguntas con las compañeras y docente de la materia, se utilizó la aplicación, Jamboard, pizarra en la nube.

Formulación de la hipótesis

La primera tarea de esta actividad consistió en que el estudiante formule su propia hipótesis. Cada estudiante formuló su propia hipótesis.

2.4 Objetivos – Saberes - Producto final - Cronograma

A continuación recordamos los objetivos y saberes de aprendizaje que plantea esta actividad para los estudiantes.

Objetivos:

- Demostrar que su hipótesis es verdadera.
- Proponer una solución tecnológica al problema planteado.

- Saberes:
- Resolver situaciones problemáticas que demanden el estudio de variables pertenecientes al suelo, clima y planta, a través de una solución que invo-lucre la aplicación de recursos tecnológicos innovadores que sirvan para llevar a cabo el proceso de recolección o modelado de los datos obtenidos.

A continuación desarrollamos un cronograma de las tareas previas que lo conducirán a la producción final. Utilizamos la herramienta diagramas de Gantt de Canvas, para determinar el cronograma actividades, tareas y subtareas con fechas límites.

2.5 Experimentación

Fueron varias las hipótesis propuestas para responder la pregunta planteada: ¿Cómo evitar el riego excesivo de las plantas? Y de esa manera cuidar el recurso hídrico y colaborar con la conciencia colectiva

A continuación comparto una de las hipótesis:

Hipótesis I

Las plantas no necesitan que el suelo esté lleno de agua, con que haya una pequeña cantidad de agua, será suficiente. El riego óptimo sería entonces un riego por capilaridad.

Desde el punto de vista de la física: los capilares o xilemas de una planta mientras más finitos sean, con mayor facilidad el agua va a trepar por ellos.

Si al capilar se moldea como si fuera un cilindro, ocuparían muy poco volumen de agua, porque tienen un radio muy pequeño.

Hipótesis II

Proteger la planta con un cobertor plástico, para que la cantidad de agua necesaria para regarla sea de menor cantidad gracias a la conservación de la humedad.

Una planta en un suelo óptimo absorbe agua por las raíces, esta viaja por los xilemas hasta llegar a las hojas, donde se evapora y pasa a la atmósfera. En este proceso de evapotranspiración, la mayoría del agua absorbida por la planta se evapora en las hojas. Entonces, si cubrimos la planta con un cobertor, la evaporación en las hojas se verá reducida y de esta manera se podrá conservar más humedad en la planta, por lo cual se requerirá de menos cantidad de recurso hídrico para regarla.

Las diferentes propuestas mencionaron riego por goteo, riego por exudación y riego por capilaridad.

Para probar la hipótesis, utilizamos 4 macetas del mismo tamaño, con una planta de la misma especie en cada una. A continuación le aplicaron a cada maceta un riego diferente (goteo, riego por exudación y riego por capilaridad).También a una de ellas sumamos el procedimiento propuesto para mantener la humedad en la planta (cubrirla en las noches con un cobertor)

a) Mediciones

La siguiente tarea que realizaron fue realizar las mediciones correspondientes, es aquí donde se introdujo la placa Arduino y un sensor de humedad, para poder determinar la humedad obtenida cada día con el riego.

Uno de los grupos sumó también la iniciativa de tapar la planta en las noches, porque en principio evitaría la evapotranspiración. Cabe aclarar que solo taparon el envase durante 8hs, durante la noche, es decir de 00:00 a 8:00

De esta manera, esta última hipótesis sostenía que el agua requerida para hidratar la planta será incluso mucho menor que lo estimado con cualquiera de los riegos propuestos (goteo, exudación y capilaridad)

b) Creando código con Arduino

En la cuarta tarea desarrollaron el código o programa en Arduino correspondiente para llevar a cabo la lectura del testeo de humedad

c) Registro y evaluación de mediciones

En la quinta tarea los estudiantes registraron las lecturas, compararon las mediciones efectuadas en cada momento del día (mañana, tarde, noche). Luego armaron una tabla para almacenar el registro de todas las mediciones efectuadas en cada momento del día, durante al menos una semana.

Por otro lado, los estudiantes acompañaron el registro de las mediciones con fotos y anotaciones escritas en su cuaderno de campo o agenda online, en ellas volcaron cada una de las mediciones efectuadas. También acompañaron con observaciones y fotos sobre el posible cambio en el aspecto de la planta

2.6 Emisión de juicios y conclusiones, redacción de un informe en un procesador científico

Los estudiantes reflexionaron sobre las mediciones efectuadas, y sobre sus hallazgos. Realizaron observaciones, comparaciones, analogías, clasificaciones y finalmente emitieron juicios y conclusiones, las cuales volcaron a un informe.

Retos científicos y tecnológicos

Finalmente se propusieron varios retos o desafíos a los estudiantes, tanto científicos como tecnológicos. Se resume a continuación dichos retos

Retos científicos (incorporando otras variables atmosféricas)

- Incorpore también el cálculo de la Humedad Relativa a las mediciones efectuadas
- Mida en días de alta temperatura (T) ¿Se observan grandes diferencias en la variable a medir? Regístrelas
- Mida en días de mucho viento ¿Se observan grandes diferencias en la variable a medir? Regístrelas
- Mida en días de Mucho o poco sol, radiación solar (RS)

Retos tecnológicos

- Imagine que debe viajar o por alguna circunstancia no podrá regar sus plantas durante un tiempo. Desarrolle un programa en Python que determine la cantidad

de agua necesaria para mantener hidratada una planta durante una X cantidad de días. Tenga en cuenta todas las variables necesarias para resolver este desafío. Como regla general, proporcione agua a sus plantas hasta un 5-10% de la capacidad de la maceta

2.7 Producto final – Reflexión - Divulgación

Finalmente los estudiantes desarrollaron un resumen científico con Látex [7], con lo aprendido en las indagaciones efectuadas en la actividad propuesta. Sumaron a este artículo científico, las respuestas a las preguntas formuladas en desafíos, también se incorporaron los resultados obtenidos en los retos, tanto los científicos como los tecnológicos. Esta documentación servirá a futuros estudiantes abordar las conclusiones obtenidas.

Para la presentación final, utilizaron un presentador de diapositivas como Google Slides, con su plantilla proyecto de ciencias de Google para llevar a cabo la exposición y discurso del trabajo de investigación efectuado. Se eligió esa plantilla porque permite explicar un proyecto de ciencias, desde la hipótesis inicial hasta la exposición de los datos del experimento, conclusiones obtenidas y producto final logrados.

3. Resultados

Después de llevar a cabo la experiencia propuesta, los estudiantes pudieron comprobar que el método de riego ideal para mantener hidratada una planta es el riego por capilaridad. Sin embargo si sumamos la técnica de tapar la planta por las noches con un cobertor, la planta mantendrá su humedad y requerirá menos cantidad del recurso hídrico para riego.

Los recursos tecnológicos utilizados, placas arduino, sensores, facilitaron el acceso a los datos experimentales, enriquecieron su análisis y favorecieron los tiempos de recolección de datos, ya que los estudiantes pudieron invertir menos tiempo en el proceso de toma de datos e invertir más tiempo en el análisis y la interpretación de los datos obtenidos. También facilita la comprensión matemática de los cálculos estadísticos posibles.

4. Conclusiones

Estas prácticas resultaron beneficiosas para los estudiantes, para su formación científica. El uso de las plaquitas arduino para llevar a cabo el testeo de humedad correspondiente y la manipulación en sí de materiales concretos permitieron una mejor comprensión y dominio cognitivo del alumno.

Las actividades STEAM permitieron también al alumno transitar por cada uno de las fases propias del método científico y comprendieron la relevancia de trabajar con herramientas que faciliten el acceso a los datos experimentales, enriquezcan su análisis y favorezcan los tiempos de recolección de datos.

Además los estudiantes pudieron inferir que las experiencias realizadas resultaron ser análogas a las prácticas reales que lleva a cabo un investigador.

También se pudo observar un mejor aprendizaje y rendimiento de los estudiantes en las materias involucradas o integradas en la actividad propuesta.

Finalmente se observó una gran predisposición y entusiasmo para desarrollar la actividad, no solo en la fase de experimentación, también para llevar a cabo o resolver los desafíos tecnológicos y científicos propuestos.

Referencias

1. Aguilera, D; Morales, D; Lupiáñez, José; Vílchez González, J; Perales, F. (2021). Objetivos de la educación STEM. Revisión Sistemática. XI Congreso Internacio-nal sobre Investigación en la Didáctica de las Ciencias. 1939-1941. Lisboa: Enseñanza de las Ciencias. Recuperado de: https://congresoenseciencias.org/actas/
2. Competencias del siglo XXI en proyectos co-tecnocreativos https://www.researchgate.net/publication/323549661_Competencias_del_siglo_XXI_en_proyectos_co-tecnocreativos
3. Sánchez Ludeña, E. La educación STEAM y la cultura «maker». Padres Y Maes-tros / Journal of Parents and Teachers, (379), 45-51(2019).
4. Domènech Casal, J. STEM: Oportunidades y retos desde la Enseñanza de las Ciencias, Universitas Tarraconensis. Revista de Ciencias de la Educación, 155-168 (2019)
5. Domènech-Casal, J. Aprendizaje Basado en Proyectos en el marco STEM. Componentes didácticos para la Competencia Científica. Ápice. Revista de Educación Científica, 2(2), 29-42 (2018).
6. Osorio U., Alfonso (2012) Mecanismos de control de variables que inciden en el riego (Clima, suelo, planta) [en línea]. Paraguay: IICA. Disponible en: https://hdl.handle.net/20.500.14001/33381 (Consultado: 25 septiembre 2022).
7. Overleaf. (s.f.). Overleaf. Recuperado el 30 de Mayo de 2023, de https://es.overleaf.com/

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Incorporación a un serious games de un proceso evaluativo basado en analíticas de aprendizaje y lógica difusa

Leonel Guccione[1], Stella Massa[1], Guillermo Lazzurri[2], Adolfo Spinelli[1], Franco Kühn[1]

[1] Grupo de Investigación en Tecnologías Interactivas, Facultad de Ingeniería, Universidad Nacional de Mar del Plata. Juan B. Justo 4302.

{leonel, smassa, spinelliadolfo}@fi.mdp.edu.ar, fkuhn@mdp.edu.ar

[2] Grupo de Investigación en Inteligencia Artificial, Facultad de Ingeniería, Universidad Nacional de Mar del Plata. Juan B. Justo 4302.

lazzurrig@fi.mdp.edu.ar

Abstract. En este artículo se presenta una metodología para incorporar a un serious games un proceso evaluativo basado en analíticas de aprendizaje y lógica difusa. El objetivo es contribuir al avance en la automatización del proceso de enseñanza/aprendizaje proponiendo una forma de implementar la automatización de la evaluación de las competencias académicas demostradas por el estudiante/jugador durante una sesión completa del juego. Para ilustrar la metodología propuesta se describe un trabajo de campo sobre un SG previamente creado.

Keywords: serious games, lógica difusa, analíticas de aprendizaje, rúbricas de aprendizaje.

1. Introducción

Un Serious Game es un videojuego cuyo objetivo supera al del mero entretenimiento y pretende incorporar un nuevo conocimiento en el jugador [1]. Los Serious Games aportan el atractivo lúdico al proceso de aprendizaje. Captan toda la atención del jugador sumergiéndolo en un ambiente con reglas propias [2]. Tiene como objetivo mejorar los procesos de aprendizaje aportando herramientas novedosas, efectivas y atractivas [3].

Las mediciones obtenidas mediante las Analíticas de Aprendizaje aplicadas a Serious Games permiten obtener datos que son de utilidad para observar las competencias demostradas por un jugador en una experiencia de juego [4], pero no son suficientes para completar el proceso evaluativo de una forma automatizada. El uso de las Lógicas Difusas en un proceso de evaluación de desempeño de competencias es altamente aplicable. Esto se debe a que la componente subjetiva e imprecisa de todo proceso evaluativo se combina con la capacidad intrínseca de las Lógicas Difusas de manejar la vaguedad o incertidumbre [5].

Se propone en este trabajo una forma de incorporación a un SG de un proceso evaluativo del nivel de desempeño logrado por el jugador, utilizando Lógicas Difusas a partir de los datos recolectados por las Analíticas de Aprendizaje.

2. Incorporación del proceso evaluativo al proceso de desarrollo de un SG

No es clara aún la forma en que el estudiante incorpora conocimientos a partir de una experiencia de juego, ni la evaluación del conocimiento que ha podido alcanzar sin la utilización de un cuestionario explícito [6]. Las dificultades en la medición de los resultados de aprendizaje logrados son una barrera que dificulta la adopción de los SG en la educación [7] [8].

Serrano-Laguna et al. [9] proponen una metodología para deducir los resultados del aprendizaje en una experiencia de juego utilizando un seguimiento no disruptivo, o sea, sin interrumpir la dinámica natural del SG. Esta metodología tiene en cuenta las etapas de diseño, implementación, validación y despliegue, así como también los objetivos de aprendizaje, la población objetivo, y una evaluación final del SG sobre la población elegida.

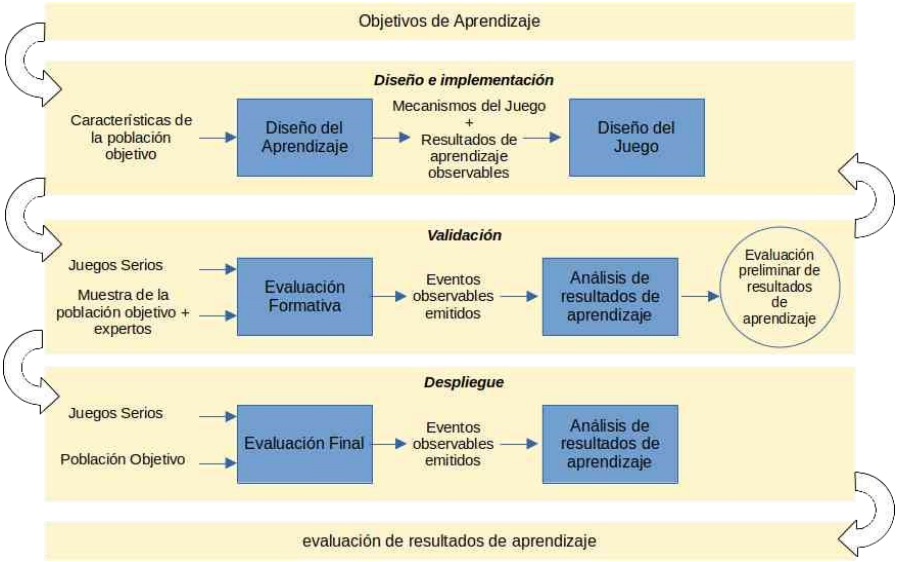


Figura 1: Proceso de diseño e implementación del SG y evaluación de resultados de aprendizaje. Adaptado de [9].

Considerando lo anterior se presenta una metodología para la incorporación de un proceso evaluativo automático a un SG, utilizando varias herramientas que actualmente se aplican a la educación, juegos y toma de decisiones a partir de datos difusos.

Este trabajo sigue el Modelo de Proceso de Desarrollo para Serious Game (MPDSG) presentado por Evans, F., et al. [10] y continuado por Spinelli, A., Massa, S., Rico, C., y Kühn, F. [11]. Se incorpora a este modelo el proceso de evaluación automática.

2.1 Proceso propuesto para el diseño e implementación de un sistema de evaluación automático

La principal diferencia de diseño entre los SG y los videojuegos está determinada en la especificación de los requerimientos que se deben considerar. Deben incluirse: contenidos y objetivos educativos junto con la especificación de la evaluación del aprendizaje y calidad educativas [12].

Los pasos a tener en cuenta para la incorporación de un sistema evaluativo a un SG son:

1. Establecimiento de los Criterios de evaluación de las competencias demostradas en la experiencia de juego utilizando las Rúbricas de Aprendizaje.
2. Determinación de lo datos observables resultantes de la interacción del jugador con el SG durante la experiencia de juego utilizando Analíticas de Aprendizaje.
3. Evaluación del desempeño alcanzado en una sesión de juegos para cada Criterio utilizando lógicas difusas.
4. Generación de un reporte de resultados a partir de la evaluación de desempeño.

La información requerida para el desarrollo del sistema evaluativo debe previamente incorporarse al documento de requerimientos del SG, por lo que, a veces será necesario volver a entrevistar a los stakeholders relacionados con la parte correspondiente (docentes en este caso).

2.2 Rúbrica de Aprendizaje

Una rúbrica es un instrumento utilizado básicamente para definir criterios de realización de tareas de aprendizaje y evaluación de competencias [13]. Permite establecer una relación entre un objeto cualitativo (texto) con objetos cuantitativos (unidades métricas). La rúbrica establece una escala de calificación clara y detallada que describe lo que se pretende del estudiante en cada tarea. Constituye un conjunto de criterios explícitos que se utilizan para evaluar el desempeño de los estudiantes [14].

Se considera el tipo de rúbrica denominado analítico. Las rúbricas analíticas descomponen los criterios de evaluación en elementos específicos y les asignan una puntuación individual. Por otro lado, las rúbricas holísticas evalúan el rendimiento general del estudiante en función de los criterios establecidos.

La creación de una rúbrica analítica involucra los siguientes pasos:

1. Definición de los objetivos de la tarea o proyecto [15].
2. Identificación de los criterios de evaluación [16].
3. Establecimiento de los niveles de desempeño o de logro que reflejen el grado en que se cumplen los criterios. Deben ser descriptivos, claros y coherentes con los criterios de evaluación [14] .
4. Descripción de los niveles de logro: Son las descripciones detalladas de lo que se espera en cada nivel de logro.
5. Asignación de puntuaciones a los niveles de desempeño que reflejen claramente el nivel de logro en relación con los criterios de evaluación [15].

En la Tabla 1 se muestra una forma de organizar la información contenida en una rúbrica.

Tabla 1: Matriz de rúbrica

Criterios de Evaluación	Nivel de Logro 1	...	Nivel de Logro j
	Calificación 1	...	Calificación j
Criterio 1	Descripción 1	...	Descripción 1j
...	...	...	...
Criterio i	...	...	Descripción ij

2.3 Analíticas de Aprendizaje

Para evaluar el aprendizaje y la calidad educativa es necesario obtener información de cada interacción relevante del jugador, o sea, aquella que refleje decisiones que involucren alguna competencia o conocimiento relacionado con el tema de estudio del SG.

Existen para esto las Analíticas de Aprendizaje (Learning Analytics), que se utilizan para el desarrollo de métricas y métodos que permiten obtener y analizar series de datos en actividades educativas [17].

La utilización de LA aplicadas a los SG se denominan Game Learning Analytics (GLA), las cuales ya están incorporadas al MPDSG. La recolección de los datos observables (que definen las GLA) se hace a través de la programación de eventos dentro del SG cada vez que el jugador realiza una interacción no trivial.

De acuerdo con los criterios especificados en las rúbricas se determinan todas las variables o datos observables que deben tenerse en cuenta para la incorporación del proceso de evaluación. Esto constituye las GLA.

En la Figura 2 se muestra la relación entre los diferentes documentos de información entre sí y con los módulos de software.

La implementación de las GLA se realiza sobre el mismo código del SG, por lo cual puede necesario intervenirlo.

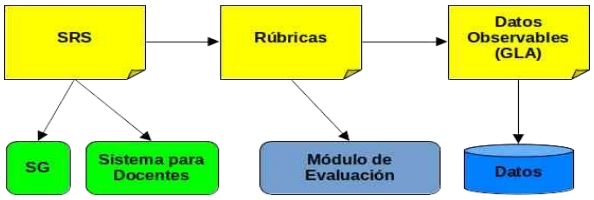


Figura 2: Relación entre los diferentes documentos y los módulos de software

2.4 Lógica Difusa

La lógica difusa pertenece al conjunto de herramientas de la Inteligencia Computacional (IC) y permite modelar y representar información no precisa, propia de muchos problemas del mundo real [18].

Permite una forma de representación del conocimiento a través de descripciones que utilizan un lenguaje cuasi natural fácil de entender. Relaciona los datos de entrada

que definen un estado del dominio representado con los datos de salida, que constituyen los resultados esperados.

Se denomina Sistema de Inferencia Difuso (FIS) [19] a aquel que es utilizado para realizar inferencias y tomar decisiones en un contexto con información imprecisa. Se forma con tres componentes principales: una base de reglas difusas, un motor de inferencia y una base de datos de conocimiento. A esto se suman los mecanismos de fuzzificación y defuzzificación (ver Figura 3).

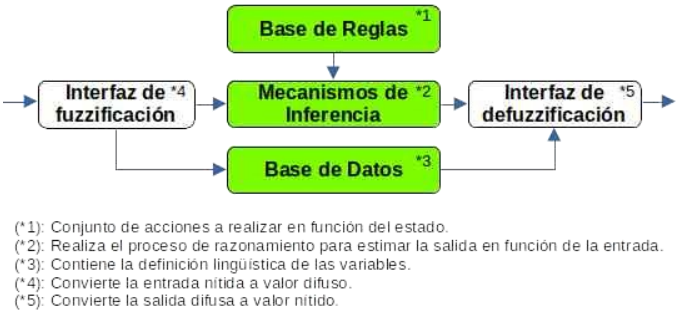


Figura 3: Estructura básica de un sistema de inferencia difuso. Adaptado de Martinez Rojas y Andrade Sosa [19]

A partir de las rúbricas se establecen las reglas de inferencia del FIS. Se utilizan los datos arrojados por las analíticas de aprendizaje como valores de entrada. Como valores de salida se obtiene la calificación correspondiente (ver Figura 4).

2.5 Componentes del sistema

Se utilizan los siguientes módulos: Serious Games, Sistema para Docente, Base de Datos, Módulo de Evaluación (ver Figura 4).

A medida que el jugador interactúa con el SG se van almacenando los valores observables (GLA) en el servidor de datos (Datos). Al finalizar el juego, el módulo de evaluación procesa todos los valores y determina una calificación.

Como resultado del proceso evaluativo, el sistema arroja un reporte sobre cada criterio observado con la calificación lograda en la experiencia de juego.

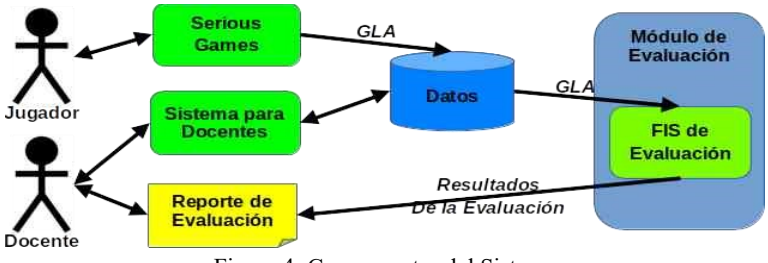


Figura 4: Componentes del Sistema

2.5.1 Experiencia de campo sobre el juego Power Down The Zombies (PDTZ)

PDTZ es un SG desarrollado por el ingeniero Franco Kühn durante su última etapa de estudiante, como parte de su Trabajo Final [4] de la carrera de Ingeniería en Informática de la UNMDP. PDTZ está enmarcado en el actual Proyecto de Investigación "Modelos de Desarrollo de Serious Games. Las Analíticas de Aprendizaje e Inteligencia Artificial" del Grupo de Investigación en Tecnologías Interactivas (GTI) de la Facultad de Ingeniería de la Universidad Nacional de Mar del Plata. El juego ha sido desarrollado siguiendo el MPDSG [10].

El objetivo académico de este juego es que el estudiante aplique y refuerce las competencias relacionadas con la utilización racional de los recursos energéticos, que corresponde a contenidos relacionados con Física y Química del cuarto año de escuelas secundarias de Argentina. El juego se desarrolla durante 10 días simulados.

2.5.2 Incorporación del sistema de evaluación automático al SG PDTZ

La incorporación del sistema de evaluación automático al SG PDTZ corresponde con la implementación del proceso descripto en el apartado 2.1 Proceso propuesto para el diseño e implementación de un sistema de evaluación automático.

En la Tabla 2 se observa la rúbrica analítica resultante de acuerdo a los pasos indicados en el apartado 2.2 Rúbrica de Aprendizaje aplicada al Criterio 1.

Tabla 2: Rúbricas para el juego PDTZ

Criterios	Alto (100%)	Medio (50%)	Bajo (0%)
C1. Comparación de Fuentes de Energía (eólica, carbón, nafta).	Ha realizado una elección MUY BUENA de generadores de energía la mayoría de los días que ha sobrevivido y ninguna elección MALA. Debe sobrevivir al menos la mitad de los días de la sesión.	Medio: Ha realizado una elección BUENA la mayoría de los días que ha sobrevivido y ninguna elección MALA. Debe sobrevivir al menos la mitad de los días de la sesión.	Ha realizado al menos una elección MALA en algún día de la sesión.

Las calificaciones MUY BUENA, BUENA y MALA para el Criterio 1 corresponden a la elección realizada de acuerdo a los recursos disponibles. También se tiene en cuenta el grado de dificultad que se presenta en cada día, que puede ser ALTO, MEDIO o BAJO.

Los componentes de una rúbrica presentan una asociación directa con las variables difusas y los términos difusos. Por lo tanto, éstas (las rúbricas) pueden servir de guía para elaborar el FIS de Evaluación.

Se muestra a continuación el diseño de las reglas para la evaluación del Criterio 1 para cada día.

Tabla 3: Evaluación del Criterio 1 para un día

Valores de entrada		Valores de salida
GD	**ELECCIÓN**	**EVALUACIÓN_DIARIA**
ALTO	MUY BUENA	SOBRESALIENTE
	BUENA	APROBADO
	MALA	REPROBADO
MEDIO	MUY BUENA	APROBADO
	BUENA	MEDIOCRE
	MALA	REPROBADO
BAJO	MUY BUENA	MEDIOCRE
	BUENA	MEDIOCRE
	MALA	REPROBADO

Luego de realizar la evaluación para cada día, se realiza la evaluación de toda la sesión del SG. Se establece otro conjunto de reglas para la calificación (ver Tabla 4).

Tabla 4: Reglas de calificación de la Sesión completa

CALIFICACIÓN	**Regla**
PROMUEVE	Si la mayoría de los días obtuvo una evaluación SOBRESALIENTE y el resto APROBADO
REPASA	Si la mayoría de los días obtuvo una evaluación APROBADO y en el resto de los días no obtuvo ninguna evaluación REPROBADO.
DESAPRUEBA	Si la mayoría de los días obtuvo una evaluación MEDIOCRE
DESAPRUEBA	Si obtuvo uno o más REPROBADO

Para la implementación del módulo de evaluación se utiliza el lenguaje Java y la herramienta QfuzzyLite [20] que genera código a partir de una interfaz gráfica de diseño de FIS.

3. Conclusiones y Trabajos futuros

El trabajo presentado describe la forma de incorporar a un SG un módulo de evaluación automatizado. La utilización de la rúbrica de aprendizaje permite una definición clara de los componentes del FIS de evaluación para cada uno de los criterios correspondientes a las competencias académicas consideradas en el SG.

El diseño adoptado posibilita que el docente determine una configuración inicial de los parámetros del juego para variar el nivel de dificultad de cada nueva sesión de juego, los cuales se almacenan en la base de datos. Esto hace que el sistema pueda fácilmente escalar hacia una etapa de mayor automatización que incorpore la capacidad de brindar recomendaciones al estudiante según el resultado de la evaluación de la sesión finalizada. Dichas recomendaciones se establecerían dentro del universo de SG en forma de un cambio en la configuración inicial de los escenarios del juego (recursos disponibles) de la siguiente sesión.

Bibliografía

1. Massa, S., Evans,F., Spinelli, A., Zapirain, E., Morcela, A., Kühn, F.: Modelos y herramientas para el proceso de desarrollo de Serious Games.XX WICC (2018)
2. Clark, C.: Serious Games.New York Viking Press (1970)
3. De Gloria, A., Belloti, F., Berta, R., Lavagnino, E: Serious Games for education and training.Internacional Journal of Serious Games. ISSN: 2384-8766 (2014)
4. Kühn, F: Juegos Serios y Analíticas de Aprendizaje: Implementación en el entorno educativo.RINFI. http://rinfi.fi.mdp.edu.ar/xmlui/handle/123456789/246 (2019)
5. Huapaya C. Lizarralde F., Gonzalez M., Arona G.: Modelo basado en Lógica Difusa para el Diagnóstico Cognitivo del Estudiante.Form. Univ. vol.5 (2012)
6. Guccione, L., Massa, S., Kühn, F.: Retroalimentación de Desempeño Basada en las Analíticas de Aprendizaje Aplicadas a Serious Games.CONAIISI 2022 (2022)
7. Alvarez, J., Michaud, L.: SG – Advergaming, edugaming, training and more.IDATE Consulting and Research (2008)
8. de Freitas, S., Liarokapis, F.: Serious games: A New Paradigm for Education?. Springer (2011)
9. Serrano-Laguna, Á., Martínez-Ortiz, I., Haag, J., Regan, D., Johnson, A., & FernándezManjón, B.: Applying standards to systematize learning analytics in serious games.Computer Standard & Interfaces, 50, 116-123. DOI: 10.1016/j.csi.2016.09.014 (2017)
10. Evans, F., Spinelli, A., Zapirain, E., Massa, S. M. Soriano, F.: Diseño centrado en el usuario, jugabilidad e inmersión.CAEDI 2016 (2016)
11. Spinelli.A., Massa, S., Rico,C., & Kühn, F.: Diseño de Serious Games. Requerimientos del Juego – Competencias y Habilidades. (2018)
12. da Rocha, R., Bittencourt, I. & Isotani, S.: Análise, Projeto, Desenvolvimento e Avaliação de Jogos Sérios e Afins: uma revisão de desafios e oportunidades. In Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE) (Vol. 26, No. 1, p. 692) (2015)
13. Alsina Masmitjà, J., Argila Irurita, A., Aróztegui Trenchs, M., Arroyo Cañada, F., BadiaMiró, M., Carreras Marín, A., Colomer i Busquets, M., Gracenea Zugarramurdi, M., Halbaut, L., Juárez Vives, P., Llorente Galera, F., Marzo Ruiz, L., Mato Ferré, M., Pastor Durán, X., Peiró Martínez, F., Sabariego Puig, M., Vila Merino, B.: Rúbricas para la evaluación de competencias. (2013)
14. Arter, J., & McTighe, J.: Scoring rubrics in the classroom: Using performance criteria for assessing and improving student performance. Corwin Press (2001)
15. Brookhart, S.: Using Performance Task-GRASPS to Assess Student Performance in Higher Education Courses. American Journal of Educational Research. 2017; 5(5):552-558. doi: 10.12691/education-5-5-12. (2013)
16. Moskal, B., Leydens, J., Pavelich, M.: Validity, Reliability and the Assessment of Engineering Education. Journal of Engineering Education. 91. 10.1002/j.21689830.2002.tb00714.x. (2002)
17. Dietz-Uhler, B., Hurn, J.: Using learning analytics to predict (and improve) student success: A faculty perspective. Journal of interactive online learning, 12(1), 17-26. (2013)
18. Jang, J. S. R., Sun, C. T., & Mizutani, E.: Neuro-fuzzy and soft computing: A computational approach to learning and machine intelligence. Prentice Hall (1997)
19. Martinez Rojas, N. & Andrade Sosa, H.: Integración de la Lógica Difusa a la Dinámica de Sistemas para la selección de terrenos de cultivos agrícolas. ELEMENTOS. 6.149.10.15765/e.v6i6.842. (2016)
20. Rada-Vilela, J., The FuzzyLite Libraries for Fuzzy Logic Control. (2018)

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Herramienta Práctica de Aplicación de Emparrillado en la Asignatura Inteligencia Artificial

Federico Ilan Maleh[1], Cinthia Vegega[1], Pablo Pytel[1], María F. Pollo-Cattaneo[1]

[1] Grupo GEMIS, Universidad Tecnológica Nacional,
Facultad Regional Buenos Aires, Argentina.
{fedemaleh, cinthia.vegega, ppytel, flo.pollo}@gmail.com

Resumen. En el presente trabajo se propone la creación de una herramienta informática que implementa una técnica de Educción de Conocimientos, utilizada en la Ingeniería del Conocimiento, llamada Emparrillado. Esta herramienta desarrollada es un recurso educativo para los estudiantes que transitan la asignatura Inteligencia Artificial de la carrera de Ingeniería en Sistemas de Información de la Universidad Tecnológica Nacional, Facultad Regional Buenos Aires, ya que muestra elementos explicativos que describen las distintas operaciones realizadas en forma simple y visual.

Palabras Clave: Inteligencia Artificial, Ingeniería del Conocimiento, Emparrillado, Repertory Grid, Educción de Conocimientos.

1. Introducción

La Ingeniería del Conocimiento [1] es una disciplina que se encuentra vinculada con la Inteligencia Artificial y que se orienta a la construcción de Sistemas Inteligentes, que son artefactos que presentan algún comportamiento inteligente en el sentido humano. Dentro de los Sistemas Inteligentes se encuentran los Sistemas Basados en Conocimiento y los Sistemas Expertos, cuya fuente de conocimiento puede proceder de los datos o del conocimiento del dominio, tal como el conocimiento de expertos humanos [2]. Para poder obtener conocimiento de un experto, se utilizan técnicas de Educción de Conocimientos, como el Análisis de Protocolos o el Emparrillado [3].

Dado esto, dentro del ámbito de la Facultad Regional Buenos Aires de la Universidad Tecnológica Nacional, la asignatura Inteligencia Artificial dictada en el quinto año de la carrera de Ingeniería en Sistemas de Información tiene como uno de sus objetivos "intervenir en el desarrollo de Sistemas Basados en Conocimiento y Sistemas Expertos" [4]. Es a partir de este objetivo, que los estudiantes que transitan esta materia profundizan y aplican en casos reales, técnicas de Educción de Conocimientos.

En este contexto, el grupo de investigación GEMIS [5] (del cual forman parte los docentes y ayudantes de la cátedra), ha detectado la necesidad de ofrecer a los estudiantes de la asignatura Inteligencia Artificial, una herramienta informática que permita aplicar la técnica de Emparrillado, siendo el objetivo de este trabajo presentar la propuesta desarrollada en esta asignatura. Para ello, en primera instancia se describe la técnica (sección 2), identificando las problemáticas que se presentan al aplicarla (sección 3). En segunda instancia, se describe la propuesta de una aplicación que permite automatizar parte de la técnica (sección 4). Finalmente, se presentan las conclusiones y futuras líneas de trabajo (sección 5).

2. Técnica de Emparrillado

La técnica de Emparrillado [6] es definida por el psicólogo estadounidense George Kelly, en el marco de su teoría de los Constructos Personales. El objetivo de la técnica es lograr una representación objetiva de los datos suministrados por una persona. El proceso de la técnica busca medir la semejanza y diferencia de los distintos conceptos identificados por la persona. En particular, en la Inteligencia Artificial se utiliza para educir conocimientos de un experto humano.

Este proceso se encuentra compuesto por 5 etapas. La primera es conocida como "Identificación de los Elementos". Durante esta etapa el Ingeniero del Conocimiento debe trabajar junto al experto en la identificación de un conjunto homogéneo y representativo de la categoría de conocimientos a analizar. La segunda etapa, llamada "Identificación de las Características", parte de los elementos identificados en la etapa anterior y consiste en listar características aplicables a dichos elementos. Estas características deben ser bipolares, es decir, tienen dos valores posibles. La siguiente etapa es el "Diseño de la Parrilla" en la que el experto debe indicar para cada elemento, que tan aplicable es cada característica. Como resultado de esta etapa, se obtiene una matriz bidimensional, donde cada valor representa el nivel de satisfacción de un elemento dado respecto a una característica. La cuarta etapa es conocida como "Formalización", es en esta etapa en la que el Ingeniero del Conocimiento procesa la información provista por el experto mediante una serie de operaciones matemáticas sobre la parrilla generada en la etapa anterior. El objetivo de esta etapa es descubrir relaciones entre los elementos y características, siendo el resultado de la misma, la generación de los Árboles Ordenados de Elementos y Características, que muestran visualmente el nivel de similitud que tienen los distintos conceptos. La última etapa es el "Análisis de Resultados", en la cual se realiza un análisis de los árboles para identificar grupos de conceptos relacionados y su similitud.

3. Identificación del Problema

La técnica de Emparrillado fue evaluada desde la perspectiva de dos posibles usuarios: el Ingeniero del Conocimiento y el estudiante de la asignatura Inteligencia Artificial. En primer lugar, el objetivo del Ingeniero del Conocimiento (al utilizar la técnica de Emparrillado) es poder educir conocimiento de un experto humano y obtener información que pueda ser incorporada a un modelo de Inteligencia Artificial que resuelva un problema dado. Desde su perspectiva, encuentra a la fase de Formalización como un proceso manual, repetitivo y propenso a errores. La cantidad de operaciones matemáticas necesarias para llevar a cabo dicha etapa, hace que consuma una gran cantidad del tiempo del ingeniero. La automatización de esta etapa permitiría utilizar su tiempo y esfuerzo en la identificación de los conceptos junto al experto y el análisis de los resultados. En segundo lugar, en el marco de la asignatura Inteligencia Artificial, el estudiante debe interiorizarse en el uso de la técnica, tanto desde el punto de vista teórico o metodológico como en la ejecución práctica del mismo. En este caso, la fase de Formalización se presenta como la de mayor complejidad para los estudiantes por diversos motivos:

- La complejidad matemática de la técnica, que requiere la realización de una gran cantidad de operaciones sobre varias matrices. Al realizar en forma manual estas operaciones, la probabilidad de un error es alta y la identificación de este, se presenta dificultosa.
- La bibliografía existente de la técnica está expresada en términos de fórmulas matemáticas escuetas, que complejizan el análisis y razonamiento del estudiante.
- Hasta la actualidad no existe una herramienta informática práctica que muestre la aplicación de la técnica y permita al estudiante verificar su trabajo y comprender de una forma visual los distintos pasos de la ejecución de la técnica.

En vista de estos problemas, se identifica la necesidad de la creación de una herramienta informática que resuelva la fase de Formalización. Esta herramienta debe, además, proveer elementos explicativos que describan las distintas operaciones realizadas en términos simples y muestren visualmente la aplicación de la técnica.

4. Propuesta de Solución

A partir de las necesidades identificadas en la sección anterior, se define como objetivo del trabajo, la creación de una herramienta informática que permita, por un lado, al Ingeniero del Conocimiento, centrar su tiempo y esfuerzo en el proceso de Identificación de Elementos y Características y el Análisis de las Parrillas, y por otro lado, brindar a los estudiantes de la asignatura Inteligencia Artificial una herramienta que facilite la comprensión de la técnica, permitiendo validar y corregir la resolución de los ejercicios propuestos por la cátedra.

En esta sección, se presenta la aplicación desarrollada, para lo cual, en primer lugar, se realiza una descripción en detalle de la etapa de Formalización del Emparrillado (sección 4.1), en la que se describen la totalidad de las operaciones matemáticas necesarias para realizarlo. Luego, se presenta la arquitectura de la solución (sección 4.2) y se muestran las funcionalidades del producto implementando (sección 4.3). Esta aplicación se utiliza en la asignatura Inteligencia Artificial durante el primer cuatrimestre (Marzo - Julio) de 2023, con un total de 74 estudiantes. Se presentan estadísticas del uso de la aplicación:

- Emparrillados Resueltos: 138
- Issues Reportados: 12
- Issues Resueltos: 8.
- Issues Pendientes: 4

4.1 Formalización del Emparrillado

La etapa de formalización aplica operaciones matemáticas sobre la matriz (o parrilla) de Elementos y Características, resultante de aplicar las primeras tres etapas de la técnica. En esta etapa, se busca medir la similitud entre los distintos conceptos utilizados en el Emparrillado. Esta etapa se divide en dos fases [6].

4.1.1 Clasificación de Elementos

Esta subetapa busca identificar los elementos que se parecen más entre sí. El primer paso es generar la parrilla de Elementos, que es el resultado de aplicar la distancia Manhattan [7], la cual se define como la suma de las distancias entre dos vectores en cada una de sus dimensiones (fórmula 1),

$$d\ (V1,V2)\ =\ \sum_{i=1}^{n}\ \ |V1_i\ -V2_i\ | \qquad \textbf{(1)}$$

donde *V1* y *V2* son los vectores sobre los que se calcula la distancia y *n* la cardinalidad de los vectores. A continuación, se presenta un ejemplo del cálculo de la distancia Manhattan.

Como se observa en la fórmula 2, la distancia entre el Elemento 3 y el Elemento 4 se calcula,

$$\begin{aligned} &d\ (E3,E4)\ =\ \sum_{i=1}^{4}\ \ |E3_i\ -E4_i\ | \\ &=|E3_1\ -E4_1\ |\ +\ |E3_2\ -E4_2\ |\ +\ |E3_3\ -E4_3\ |\ +\ |E3_4\ -E4_4\ | \\ &=|1\ -\ 2|\ +\ |3\ -1|\ +\ |2\ -4|\ +\ |2\ -3| \\ &=|-1|\ +\ |2|\ +\ |-2|\ +\ |-1|\ =\ 6 \end{aligned}$$

En la figura 1 se presenta la visualización del ejemplo anterior en la aplicación desarrollada.

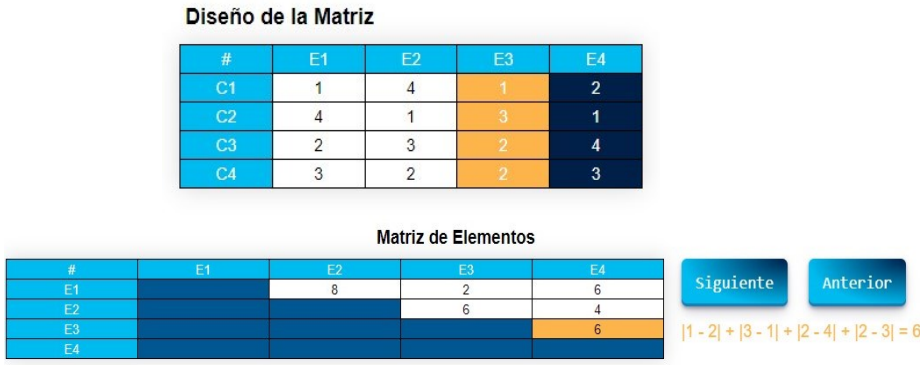


Fig. 1. Imagen de la aplicación que muestra el cálculo de la distancia Manhattan entre Elementos.

Luego, se realiza la reducción de la parrilla, unificando en cada paso los elementos que tienen la menor distancia de la parrilla. En la figura 2, se observa un ejemplo de la reducción, donde el primer paso de reducción unifica a los elementos *E1* y *E3* con una distancia de 2 y el segundo paso unificará a *E2* con *E4* a una distancia de 4.



Matriz de Elementos

#	E1	E2	E3	E4
E1		8	2	6
E2			6	4
E3				6
E4				

Clasificación de Elementos

#	E1,E3	E2	E4
E1,E3		6	6
E2			4
E4			

Distancia mínima es 4 entre E2,E4

Fig. 2. Imagen de la aplicación que muestra la reducción de la parrilla.

Finalmente, una vez reducida la parrilla, se obtiene el árbol ordenado de Elementos (figura 3), que grafica las mínimas distancias identificadas durante la etapa de reducción de la parrilla de Elementos.

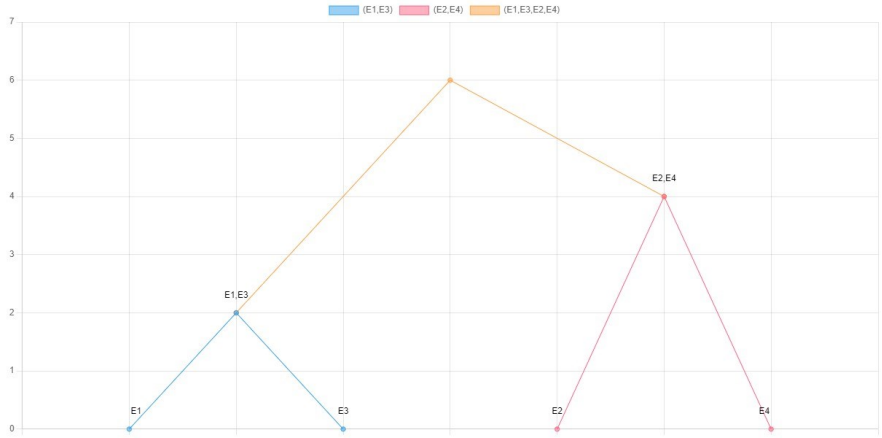


Fig. 3. Imagen de la aplicación que muestra el árbol ordenado de Elementos.

4.1.2 Clasificación de Características

En esta fase se identifica el nivel de similitud entre las características, utilizando un proceso muy similar al realizado para los elementos. Para ello, hay que generar en primer lugar algunas matrices auxiliares necesarias para este proceso.

La primera parrilla generada se denomina “Parrilla Opuesta” y es una réplica de la parrilla original, reemplazando cada valor por su opuesto. Por ejemplo, dada una matriz que utiliza los valores del intervalo [1,4], se reemplazan todos los 1 por 4, los valores 2 por 3 y viceversa. En la figura 4, se observa un ejemplo de la construcción de la Matriz Opuesta.

Luego, se genera la “Matriz Distancia de Características” que contiene las distancias entre características. El triángulo superior de la matriz se completa con las distancias Manhattan de las características, en forma similar a lo realizado con los elementos,

comparando filas en lugar de columnas. El triángulo inferior de la matriz (no utilizado para los elementos), se llena mediante la distancia Manhattan entre características, tomando los valores de la primera característica de la matriz original, mientras que los valores para la segunda característica se obtienen de la Matriz Opuesta recientemente generada, como se puede observar en la figura 5.

La última matriz por generar es la "Matriz de Mínima Distancias", que es la equivalente a la "Matriz de Elementos". Para generar esta matriz, se selecciona el menor valor para cada par de características de la Matriz Distancia (fórmula 3).

$$\text{Matriz Min Distancia I, J} = \text{Min (Matriz Distancia I, J, Matriz Distancia J, I)} \quad \textbf{(3)}$$

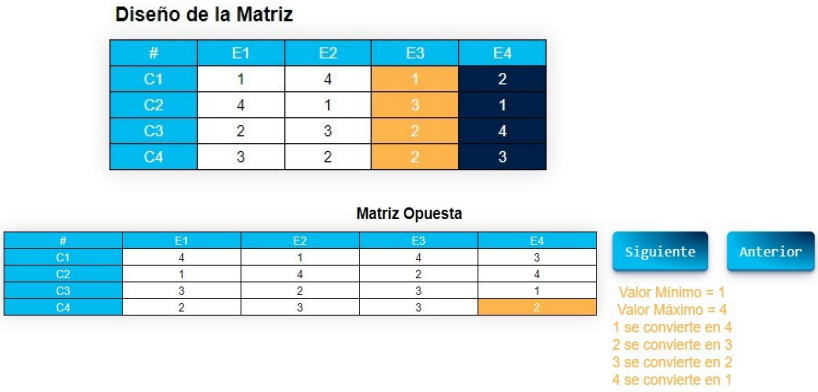


Fig. 4. Imagen de la aplicación que muestra la parrilla opuesta.



Fig. 5. Imagen de la aplicación que muestra la Matriz Distancia.

La figura 6, presenta un ejemplo del cálculo de la Matriz de Mínimas Distancia, en la que se observa que la distancia mínima entre C2 y C3 tiene como posibles valores 2 y 6, por lo que se selecciona 2 como valor mínimo.

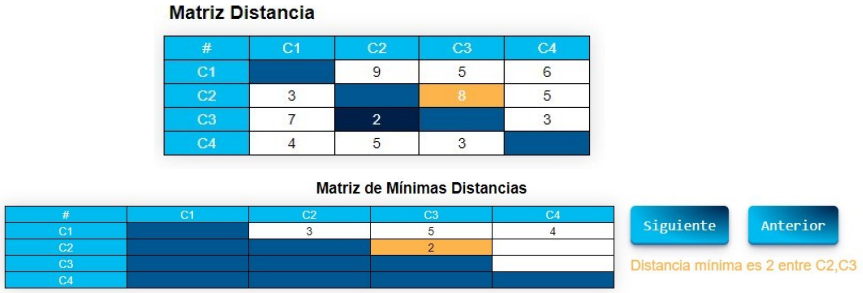


Fig. 6. Imagen de la aplicación que muestra la Matriz de Mínimas Distancias.

Finalmente, tomando la Matriz de Mínima Distancias se lleva a cabo el proceso de reducción y generación de árbol ordenado, siguiendo el mismo procedimiento realizado para los elementos. Para el caso tratado en esta sección, el árbol ordenado de características resultante se observa en la figura 7.

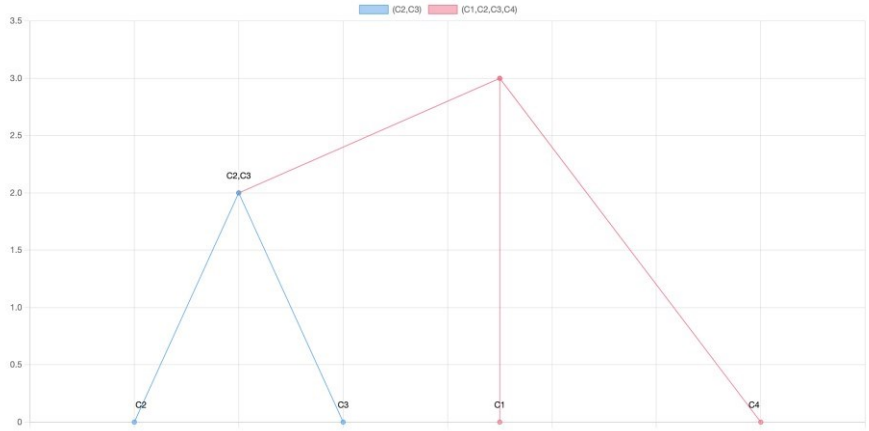


Fig. 7. Imagen de la aplicación que muestra el árbol ordenado de Características.

4.2 Arquitectura de la Solución

La presente sección describe la solución tecnológica implementada para la generación de la herramienta informática. En primer lugar, se definen los componentes de arquitectura desarrollados y el despliegue de la misma (sección 4.2.1). En la segunda parte, se definen las tecnologías utilizadas durante el desarrollo (sección 4.2.2).

4.2.1 Componentes

La solución realizada consiste en una aplicación web, compuesta por dos componentes. El primer componente, llamado "Backend Service" es un servidor REST [8]. Este componente expone un recurso que toma como input una Parrilla de

Elementos y Características y resuelve la lógica matemática de la técnica. El output del servicio incluye todas las matrices requeridas por la técnica, como una explicación de todas las operaciones matemáticas que fueron necesarias para su construcción. El segundo componente es el cliente web, la interfaz visual que permite al usuario interactuar con el servidor y poder visualizar la resolución del Emparrillado. La figura 8, muestra en forma gráfica la arquitectura descrita.

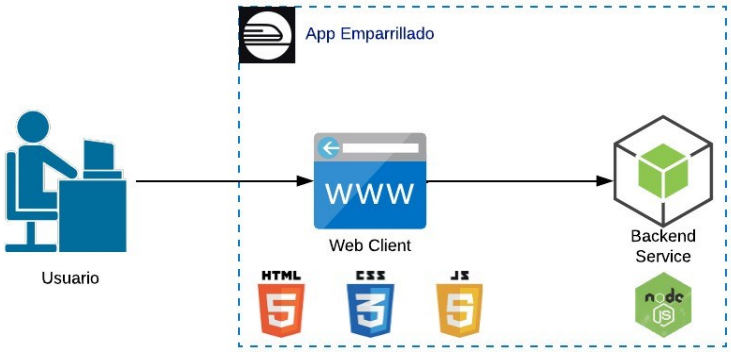


Fig. 8. Arquitectura de la Solución.

Las ventajas que provee la arquitectura presentada son:

- *Desacoplamiento.* La arquitectura desacopla la resolución de la técnica de la interfaz visual utilizada para presentarla. Esto permite en el futuro la implementación de distintas interfaces, como, por ejemplo, una aplicación mobile, sin requerir implementar nuevamente la lógica matemática.
- *Portabilidad*. La utilización de una interfaz web, permite el acceso a la aplicación de una multitud de dispositivos, desde computadoras hasta celulares, sin restricciones de marca o modelo. El único requerimiento es un navegador web y acceso a Internet.

En cuanto al despliegue de la solución, se utiliza la plataforma Railway [9], un servicio de Platform as A Service (PAAS) [10], que permite el despliegue de aplicativos en forma sencilla. Los servicios ofrecidos por la plataforma son:

- *Despliegue de Aplicaciones*. La plataforma provee el despliegue automático de aplicaciones para un conjunto de tecnologías soportadas. Únicamente es necesario brindar acceso al repositorio del código y cumplir con el formato de proyecto (template) especificado. Todas las aplicaciones desplegadas reciben un dominio web que puede ser distribuido a usuarios.
- *Despliegue de Servicios*. La plataforma ofrece servicios de Cache y Base de Datos que se pueden desplegar. Las aplicaciones acceden a los servicios mediante variables de entornos definidas por la propia plataforma.
- *Monitoreo*. Todas las aplicaciones desplegadas en Railway cuentan con un monitoreo básico de la salud de los servidores.

4.2.2 Tecnologías

Las tecnologías utilizadas para el desarrollo del "Web Client" fueron HTML, CSS y JavaScript. La solución no hace uso de frameworks JavaScript para el desarrollo de la app, de modo de simplificar el desarrollo y simplificar el stack tecnológico. La selección de estas tecnologías fue limitada por la selección de la web como interfaz de desarrollo. En cuanto al "Backend Service", se utiliza JavaScript como lenguaje de programación y Node.JS [11] como framework de desarrollo. La selección de JavaScript se debe a utilizar la menor cantidad de tecnologías (siendo JavaScript ya utilizado por el "Web Client"), mientras que la selección de Node.JS se justifica en ser uno de los principales framework de desarrollo de aplicaciones backend en JavaScript en la actualidad [12].

4.3 Funcionalidades de la Aplicación

La solución desarrollada consiste en una aplicación web en la que el usuario comienza ingresando la Parrilla de Elementos y Características.

Una vez ingresada la parrilla, la aplicación realiza la etapa de Formalización, mostrando paso a paso el armado de las parrillas definidas por la técnica. El usuario puede visualizar en cada paso las operaciones matemáticas que fueron realizadas, mostrando de que parrilla se obtuvo cada valor y mediante qué operaciones se llega a cada resultado. Finalmente, se hace uso de una herramienta de graficado para la generación de los árboles ordenados de Elementos y Características.

Otras funcionalidades incluidas en la versión actual de la aplicación incluyen (I) la posibilidad de guardar parrillas para poder reutilizarlas, (II) disponibilidad material audiovisual de la técnica, mediante un enlace a videos explicativos desarrollados por la cátedra de Inteligencia Artificial [13].

5. Conclusiones

El presente trabajo no es más que una exploración inicial de la problemática, con un enfoque mayor en el estudiante de Inteligencia Artificial y en la explicación gráfica y paso a paso de la lógica matemática de la técnica. Por lo tanto, se concluye que el desarrollo pudo ser logrado. La aplicación fue utilizada como herramienta de estudio, facilitando la comprensión de la técnica al mostrar visualmente su aplicación y como herramienta de práctica, permitiendo a los estudiantes validar sus propios resultados de los ejercicios de práctica.

Como futuras líneas de trabajo las siguientes funcionalidades se prevén incorporar:

- Permitir describir los elementos y características. Esta funcionalidad es necesaria para enmarcar la aplicación en un contexto dado.
- Permitir el guardado de las distintas parrillas, para generar una base de datos de problemas resueltos. La solución actual permite el guardado de las parrillas únicamente en el file system local del dispositivo.
- Introducir los conceptos de usuarios y equipos al sistema. Es una funcionalidad que permitiría a un equipo de Ingenieros del Conocimiento compartir un

repositorio de parrillas en forma segura (y privada).

- Introducir el concepto de "Proyecto". La técnica de Emparrillado es factible de realizarse para un mismo problema con un gran número de expertos humanos. La aplicación actualmente no permite el concepto de agrupar distintas parrillas como "soluciones" a una misma problemática y calcular estadísticas en base a los grupos y relaciones identificadas.

Referencias

1. García Martínez, R., Britos, P. (2004) Ingeniería de Sistemas Expertos, Editorial Nueva Librería.
2. Vegega, C., Pollo-Cattaneo, M. F., Olivas, J. A. (2022) A Preliminary Approach about using nowadays Knowledge Engineering in Artificial Intelligence: a literature overview. Short Papers of the 10th Conference on Cloud Computing, Emerging Topics (JCC-BD&ET 2022). Universidad Nacional de La Plata, Facultad de Informática. ISBN 978-950-34-2126-0.
3. Vegega, C., Pytel, P., Pollo-Cattaneo, M.F. (2020). Evaluation of the Bias in the Management of Patient's Appointments in a Pediatric Office. ParadigmPlus. Vol. 1, Nro. 1. Págs. 1-21. ISSN: 2711-4627.
4. Universidad Tecnológica Nacional – Facultad Regional Buenos Aires (2023). Diseño Curricular de la Carrera Ingeniería en Sistemas de Información. Plan 2023. Disponible en: https://www.frba.utn.edu.ar/wp-content/uploads/2022/12/RES-3120-22-PLAN-DE-TRSNSICION-ING.-EN-SISTEMAS-DE-INFORMACION-1.pdf. Último Acceso: 26/07/2023.
5. Grupo GEMIS (2023). Historia Grupo GEMIS. Disponible en http://grupogemis.com.ar Último Acceso: 12/07/2023.
6. Prof. Serna, Edgar. Desarrollo e Innovación en Ingeniería. Método basado en el emparrillado para evaluar los datos aplicables para entrenar algoritmos de aprendizaje automático (pp. 106-137).
7. Black, P. (2006). Manhattan distance. In Dictionary of Algorithms and Data Structures, National Institute of Standards and Technology (NIST), U.S. Department of Commerce.
8. Rest. Disponible en: https://restfulapi.net. Último Acceso: 31/07/2023
9. Railway. Disponible en: https://railway.app/. Último Acceso: 26/07/2023
10. Platform As A Service. Disponible en: https://cloud.google.com/learn/what-is-paas. Último Acceso: 31/07/2023
11. Node.JS. Disponible en: https://nodejs.org/es/about. Último Acceso: 31/07/2023
12. Los 5 mejores frameworks Full Stack (2023) Disponible en https://keepcoding.io/blog/los5-mejores-frameworks-full-stack/ Último Acceso: 05/08/2023
13. Playlist Emparillado Grupo Gemis. Disponible en: https://www.youtube.com/playlist?list=PLliFKT3EY0k1BpB2SzLqiKKe-pdy1CTNQ. Último Acceso: 31/07/2023

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

eUCCvm: Una Herramienta Educativa Integral para la Programación de Sistemas Embebidos

Cristian F. Perez-Monte[1], Gustavo Mercado[1], Carlos Taffernaberry[1], Ana Laura Diedrichs[1], Fabiana Piccoli[2,3], Mario Sebastian Tobar[1], Marcelo Ledda[1], Raúl Moralejo[1], and Rodrigo Gonzalez[1]

1 GridTICs, Universidad Tecnológica Nacional, Facultad Regional Mendoza, Argentina
2 Universidad Nacional de San Luis, Ejército de los Andes 950, San Luis Argentina
3 Universidad Autónoma de Entre Ríos, 25 de Mayo 385, Concepción del Uruguay, Entre Ríos, Argentina
{cristian.perez,gustavo.mercado,carlos.taffernaberry,sebastian.tobar, rodrigo.gonzalez}@gridtics.frm.utn.edu.ar mpiccoli@unsl.edu.ar
{ana.diedrichs,marcelo.ledda,raul.moralejo}@docentes.frm.utn.edu.ar
http://www.gridtics.frm.utn.edu.ar

Abstract. Replantear nuestras prácticas y formas de enseñanza, ya sea por acontecimientos mundiales como fue la pandemia o por particularidades de nuestra realidad: condiciones sociales, económicas, etc. es una necesidad. En este trabajo presentamos una herramienta de código abierto y compilación cruzada universal para la programación de Sistemas Embebidos: *eUCCvm*. Mediante su uso es posible desarrollar actividades de programación de aplicaciones para Sistemas Embebidos sobre múltiples arquitecturas. Puede ser utilizada en el ámbito académico como laboral, y en cualquier modalidad: virtual o presencial. Finalmente mostramos una breve encuesta sobre su inclusión en algunas cátedras de la Universidad Tecnológica Nacional, Regional Mendoza.

Keywords: *Sistemas Embebidos, Programación de Arquitecturas, Software Educativo, Código Abierto*

1. Introducción

eUCCvm (*Embedded use Universal Cross Compiler Virtual Machine*) es un sistema basado en una máquina virtual para desarrollo, compilación y programación de sistemas embebidos de uso gratuito, usada como herramienta para el desarrollo de las clases prácticas en diferentes cátedras de la Universidad Tecnológica Nacional, Facultad Regional Mendoza. Inicialmente fue desarrollada en el año 2017 para la programación específica de EDU-CIAA [4,8]. En 2018 y 2019 se extendió su uso a otras arquitecturas de microcontroladores y microprocesadores [26], aquellas utilizadas principalmente en la cátedra de Técnicas Digitales II [24]y III [25]. A partir del año 2020, con el inicio de la pandemia, se inició un proceso para la adaptación de todo el sistema educativo y la currícula por la virtualidad. De igual forma, en este mismo sentido se hicieron una serie de mejoras sobre *eUCCvm* a fin de adaptarlo, teniendo en cuenta las nuevas necesidades de alumnos y docentes para la realización de actividades de la cátedra derivadas del ASPO (Aislamiento Social Preventivo y Obligatorio dispuesto por Decreto 297/2020). En los años siguientes continuamos agregando arquitecturas de microcontroladores y microprocesadores, incluidas las usadas en computadoras hogareñas y consolas de

video juegos. Finalmente a partir de este año se agregaron los microcontroladores con arquitectura de hardware abierto RISCV [21], los cuales son centrales para el futuro del dictado de las cátedras de técnicas digitales. Además de lo expuesto, como *eUCCvm* es una herramienta pública y de uso educativo, es posible descargarla desde [23].

Este trabajo está organizado de la siguiente manera: en la sección 2 describimos su impacto en el ámbito académico. Las secciones 3 y 4 detallan las características generales y específicas respectivamente de *eUCCvm*. La sección 5 muestra resultados en relación al impacto de su aplicación. Finalmente, en la sección 6 presentamos las conclusiones y trabajos futuros.

2. *eUCCvm* y su Aplicación en el Ámbito Académico

eUCCvm está desarrollado para permitir aprender a programar sistemas embebidos en un entorno seguro y controlado, sin tener que preocuparse por dañar su propio sistema operativo.

Para el diseño y desarrollo incorporamos la perspectiva CTS (Ciencia, Tecnología y Sociedad) [20,6,19], la misma consiste en tener en cuenta los aspectos ideológicos, culturales y la experiencia personal (valores, sentimientos, motivaciones, entre otras), además de los aspectos técnicos y operativos.

eUCCvm permite utilizar computadoras no necesariamente con muchos recursos ya que demanda pocos requerimientos para su utilización por los alumnos, e ideal para en países subdesarrollados o en vías de desarrollo. Además, permite resolver problemas como los impuestos durante la virtualidad: el acceso intermitente, deficiente o nulo a Internet, constituyendo una importante barrera para la realización de actividades. En este contexto, *eUCCvm* brinda una herramienta integral donde tiene todo incluido y está lista para ser usada sin la necesidad de acceso permanente a Internet en la realización de actividades virtuales de los alumnos. Así es posible realizar actividades indistintamente en la virtualidad desde sus casas sin necesidad de recurrir a recursos computacionales avanzados o acceso a Internet, como también desde el ambiente académico, maximizando en este caso el tiempo de clase dedicado a la programación y desarrollo de proyectos con microcontroladores. En virtud de ello, para adaptarlo a las condiciones impuestas por la virtualidad, *eUCCvm* cuenta no solo con la capacidad de programar, compilar y grabar sobre el hardware de muchas arquitecturas sino que dispone de documentación de soporte y actividades desafíos para la competencia individual o entre grupos.

Otra ventaja que ofrece es su naturaleza de código abierto, la utilización de una herramienta con este tipo de código fuente abre la posibilidad a los estudiantes de modificar la herramienta para potenciarla para sus desarrollos propios, y a los docentes de adaptar sus prácticas. Asimismo se le suma la posibilidad de utilizar hardware abierto, como la arquitectura RISCV y los beneficios de su uso [12].

eUCCvm brinda un amplio abanico de plataformas de hardware de microcontroladores, no solo para fortalecer las habilidades de programación en cada uno de ellas sino también para fortalecer las capacidades de aprendizaje de nuevas arquitecturas que puedan surgir en el futuro. En otras palabras se refuerzan las capacidades de ”aprender a aprender” nuevas arquitecturas y/o lenguajes de

programación para los mismos. Además, al contar con arquitecturas de diferentes niveles de complejidad, le posibilita a los estudiantes aprender a su ritmo y aplicar sus conocimientos a medida que avanza en el curso, permitiendo desarrollar proyectos completos sobre hardware real ya sea en forma individual o grupalmente.

Otra de sus características y que consideramos un incentivo adicional, es la inclusión de arquitecturas utilizadas en la industria de los videojuegos, a fin de conocerlas y poder programarlas, no solo para ampliar el conocimiento respecto a nuevas arquitecturas, sino también de focalizarse en aquellas que los estudiantes están familiarizados por su utilización previa. Ya es ampliamente conocido y debatido que la integración de los videojuegos en el ambiente educativo es muy beneficioso [7] [11].

Las potenciales habilidades y beneficios a lograr por la utilización de *eUCCvm* en los estudiantes que realizan materias relacionadas a la programación de Sistemas Embebidos son:

- Potenciar la programación en los lenguajes ensamblador y C para diferentes microcontroladores.
- Al permitir no solo la programación, sino también la compilación y grabación del programa sobre el hardware real, se permiten desarrollar prácticos completos, con lo cual los estudiantes pueden ver cómo se aplican los conceptos teóricos en situaciones del mundo real.
- Al ofrecer la posibilidad de trabajar sobre una variedad de arquitecturas de hardware de microcontroladores, no solo fortalece las habilidades de programación en cada uno de ellas, sino también potencia las destrezas en el aprendizaje de nuevas arquitecturas que puedan surgir en el futuro.
- Impulsar habilidades en la programación de microcontroladores para el manejo de sistemas de almacenamiento, visualización, sensado y actuadores.
- Potenciar habilidades en el diseño y programación de interfases de comunicaciones para la interconexión entre microcontroladores o entre microcontroladores y otros dispositivos como sensores, actuadores o de entrada salida.
- Favorecer el trabajo colaborativo mediante la realización de actividades desafío para potenciar el desarrollo de proyectos grupales o individuales.
- Reforzar habilidades en la utilización de lenguajes de descripción de hardware.
- Discernir sobre la arquitectura más adecuada para proyectos a desarrollar.
- Discernir sobre qué lenguaje de programación es más adecuado para determinadas arquitecturas y proyectos a desarrollar.
- Reforzar habilidades en el uso de línea de comandos para el manejo avanzado de un sistema operativo Linux.
- Reforzar habilidades en la programación, compilación, versionado y utilización de repositorios mediante Linux con línea de comandos.
- Fortalecer el conocimiento avanzado teórico y de programación en arquitecturas usadas para el entretenimiento en las cuales el alumno ya está previamente familiarizado por su uso en videojuegos.

Por todo lo expuesto, se puede determinar que *eUCCvm* posee muchas propiedades que lo hacen un buen candidato a la hora de considerar un software para el desarrollo

de prácticas de programación en Sistemas Embebidos cualquiera sea su modalidad y sin mayores requerimientos de recursos.

3. Características Generales de *eUCCvm*

eUCCvm está desarrollado sobre una máquina virtual [10] basada en el sistema operativo Linux Ubuntu 18.04 LTS. Puede utilizarse en distintos sistemas operativos de máquinas reales gracias a VirtualBox [17].

Para su instalación, se usó la versión de 32 bits a fin de permitir su virtualización incluso en computadoras sin características de virtualización por hardware o instrucciones AMD64/EMT64. Además se utilizó la instalación de servidor para obtener un sistema operativo liviano y sin entorno gráfico para la virtualización en máquinas con solo 256 MB de RAM tales como las netbooks disponibles del Plan Conectar Igualdad [30].

La máquina virtual posee diversos compiladores instalados desde los repositorios oficiales de Ubuntu. Además incluye otros compilados desde el código fuente oficial de sus desarrolladores y varios SDKs de desarrollo para distintas arquitecturas.

Adicionalmente se incluye la documentación básica para el soporte de desarrollo de hardware y la programación de cada una de las arquitecturas.

A continuación se especifican con mayor precisión algunas de las características de *eUCCvm*.

4. Especificaciones Detalladas

El sistema desarrollado agrega las siguientes capacidades al sistema operativo:

- Escritura de código. Permite la generación de código en diferentes lenguajes de programación, especialmente lenguaje ensamblador [13] y C [28].
- Modificación de códigos existentes. Interpretación del funcionamiento de códigos de ejemplos existentes y su modificación para adaptarlos a la solución de nuevos problemas propuestos.
- Compilación. Generación del código máquina para diferentes arquitecturas de hardware.
- Grabación de los programas compilados en dispositivos embebidos reales.
- Simulación de microcontroladores descriptos en Verilog (NEORV32, arquitectura RISCV32).
- Acceso remoto seguro (ssh) para permitir la utilización de múltiples consolas y acceso a todo el sistema de archivos para su edición dentro o fuera de la máquina virtual.
- Documentación de soporte.
- Entornos de desarrollo con múltiples ejemplos didácticos.
- Actividades de desafíos prácticos para resolver en forma individual o grupal.

El sistema está limitado al uso de herramientas de entorno de línea de comandos (CLI) para permitir un entorno liviano y transportable, impidiendo la ejecución de herramientas que requieran de modo gráfico (GUI). Para la escritura y modificación de código se utilizan editores estándares disponibles para Linux.

4.1 Arquitecturas, Módulos y otras Tecnologías

En la actualidad, se cuenta con el soporte de varias arquitecturas, permitiendo una fácil actualización e inclusión de otras futuras.

Las arquitecturas de sistemas embebidos consideradas para compilación, tanto en ejecución *bare metal* como para su ejecución en GNU/Linux (para el caso de las más potentes), son:

- Microchip PIC [29].
- AVR (usados en Arduino. [9] NANO, UNO y MEGA).
- Xtensa LX106 (usados por ESP8266 [27] y ESP32).
- ARM Cortex M3 [5] [32] (utilizados en Blue pill y Arduino DUE).
- ARM Cortex M4 (utilizados en Black Pill y EDU CIAA) [32] – RISC RV32IMC (usados en módulos como ESP32-CX y ESP32-H2) [15] – ARM 11 de 32 bits [1] (utilizados en Raspberry PI 1).
- ARM Cortex A, A7 de 32 bits [2] [31] (utilizados en Raspberry PI 2) y A53 de 64 bits [3] [14] (utilizados en Raspberry PI 3).

Además de las arquitecturas más avanzadas para su ejecución en Linux, tales como:

- x86 estándar y x86 (amd64/emt64).
- MIPS.
- SPARC.
- ALPHA.
- 68k.
- POWERPC.
- RISCV RV64

Finalmente, y para el caso de las computadoras hogareñas y consolas de video juegos, las arquitecturas incluidas son:

- Ricoh 2A03 (Nintendo Entertainment System)
- 6507 (ATARI 2600)
- 68K (ATARI ST - AMIGA OCS - SEGA GENESIS)
- ARM (GameBoy Advance)
- Ricoh 5A22 (Super Nintendo)
- Zilog Z80 (Sega Master System)
- MOS 6502 (Atari400/800)
- Sharp LR35902 (GameBoy)
- CPU 32/64 MIPS6 III NEC VR 4300i + GPU RCP de 64 bits (Nintendo64)
- 68K+Z80 (Sega MegaDrive)

Además, agrega ejemplos para la utilización de módulos, que ofrecen funcionalidades a las arquitecturas antes mencionadas, tales como:

- Módulos de GPIO, sensado de variables analógicas y visualización numérica de 7 segmentos.
- Pantallas LCD y OLED de alta resolución.
- Módulos de conectividad: Ethernet, de radio GPRS, WiFi o Bluetooth.
- Módulos de CCDs.
- Almacenamiento con microSD.
- Módulos de posicionamiento global.

El sistema dispone también de ejemplos sencillos para la implementación de diferentes tecnologías electrónicas de entrada-salida y comunicaciones tales como:

- Uso de entradas salidas digitales de propósito general.
- Uso de entradas y salidas analógicas.
- Protocolo de comunicaciones SPI [16] e I2C. – Protocolos de comunicaciones paralelos.

Dentro de las herramientas integradas tiene las correspondiente a la grabación de la compilación en diversas arquitecturas para sistemas embebidos como:

- Grabación por puerto USB de AVR Arduino Uno y Mega.
- ARM Blue Pill, ARM Black Pill, Arduino Due y EDU-CIAA.
- ESP8266 y ESP32.
- Microchip PIC con PICkit 2 [18].

Todos los ejemplos están disponibles dentro del sistema, ordenados en carpetas, arquitectura y módulos usados.

5. Resultados Obtenidos

Actualmente el sistema está siendo usado en las cátedras de Técnicas Digitales III, desde el año 2017, y de Técnicas Digitales II, a partir del año 2018 hasta el presente año. En la tabla 1 se muestran los temas desarrollados utilizando *eUCCvm* en el área práctica de Técnicas Digitales II, año por año. Por un lado, mediante el uso de actividades guiadas durante el desarrollo de la cátedra, se indican el hardware (módulos, consolas o sistemas computacionales) con sus respectivas arquitecturas aplicadas a la cátedra. Por otro lado, además, mediante la aplicación de actividades especiales, por desafíos grupales o individuales, se muestran también actividades específicas de integración de tecnologías o módulos de funcionalidad a diferentes arquitecturas. Se puede observar como la cantidad de actividades integradas en la cátedra realizadas con *eUCCvm* se fue incrementando año a año, siendo mucho más creciente a partir del 2020 y de su adaptación a la virtualidad.

Table 1. Áreas temáticas aplicadas

Año	Arquitecturas	Funcionalidad	# Activ.
2018	EDU-CIAA (ARM CORTEX M4)	Ej de uso de firmware v2	1
2019	Microcontroladores PIC con pickit 2 Módulo PIC DM164140(16F18855) Arduino UNO(AVR Atmega328) Arduino MEGA(AVR Atmega2560) Arduino DUE (ARM CORTEX M3) Bluepill (ARM CORTEX M3) ESP8266 (Tensilica Xtensa L106) ESP32 (Tensilica Xtensa LX6) CIAA-Z3R0 (ARM CORTEX M0+)	Ejemplos de uso de firmware v3 Desafío de uso de puertos de comunicaciones con ARM	12
2020	Blackpill (ARM CORTEX M4) Nintendo (MOS 6502 /2A03) Super Nintendo(MOS 65C816/5A22) Sega Genesis (68K + Z80) Nintendo64(MIPS6 III NEC VR 4300i + GPU RCP de 64 bits)	-	23
2021	-	Desafío de aplicación de puerto de comunicaciones y sistemas de almacenamiento con sistemas AVR	25
2022	-	Desafío de aplicación de uso de comunicaciones y conversores analógico digital con ARM	27
2023	NEORV32 y ESP32-C3 (RISCV)	En desarrollo	30(...)

Respecto al uso académico, en la tabla 2 indicamos la cantidad de alumnos que usaron *eUCCvm* en cada año de cursado, la cantidad de descargas desde el servidor de la Facultad Regional Mendoza de UTN y la cantidad de versiones que se publicaron para la incorporación de mejoras. Como se puede observar, el número de descargas está vinculada a la cantidad de alumnos y versiones publicadas.

Table 2. Impacto académico

Año	Cantidad de alumnos	Cantidad de descargas	Cantidad de versiones de *eUCCvm*
2019	44	440	31
2020	43	400	9
2021	43	405	2
2022	33	187	2
2023	16	131	4 (...)

El proyecto está siendo usado, además del ambiente académico universitario, en otros ámbitos. Desde el año 2019 está compartido en forma pública por diversos medios, uno de ellos [22], independiente de las cátedras de UTN. Desde dicho sitio, cientos de descargadas muestran el creciente interés, más allá del uso académico.

Para desarrollar un análisis del impacto de *eUCCvm* en los alumnos, se realizó una encuesta a alumnos que usaron la herramienta, con distintas modalidades, durante el

cursado de algunas de las materias, y a otros que no la usaron por no estar disponible en dicho momento. La muestra utilizada fue de 40 alumnos del año 2018 hasta el 2023 de una población aproximada de 220 y de otros 20 alumnos, quienes cursaron antes del 2018 y no la utilizaron, solo le describimos para la encuesta, las características generales de *eUCCvm*.

La encuesta consistía en los siguientes dos tipos de consultas:

- Consultas generales sobre las herramientas de software usadas en la cátedra.
- Consultas específicas sobre *eUCCvm*.

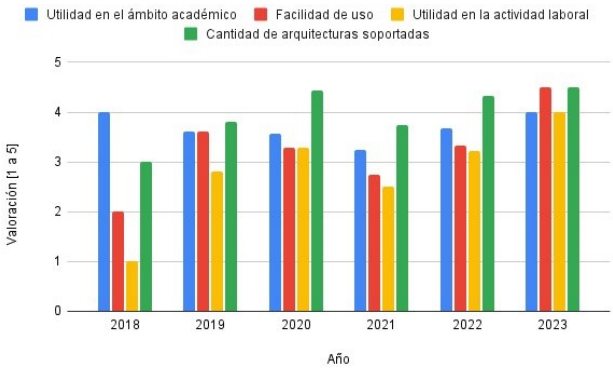


h

Fig.1. Valoraciones a las Consultas Generales de *eUCCVm*.

Respecto a las consultas generales, permitió realizar una evaluación de las herramientas por parte de los alumnos, tanto de *eUCCvm* como de otras. Esto nos permitió realizar una evaluación, incluyendo años de cursado en los cuales aún no se aplicaba *eUCCVm*.

Los resultados de las consultas generales pueden observarse en la Figura 1.



!ht

Fig.2. Consultas específicas sobre *eUCCvm*.

De las consultas específicas sobre *eUCCvm* solo pudo realizarse en forma acotada y con limitaciones a partir de alumnos del 2018 y con la mayor parte de las

funcionalidades disponibles en el año 2019. Los resultados de las consultas específicas se observar en la Figura 2.

La primer versión, 2018, sólo consideraba una placa: la edu-ciaa, a partir del 2019 se agregaron todas las detalladas en la tabla 1. En la Figura 2 se observa como los alumnos del 2018 tuvieron una valoración menor respecto a los años subsiguientes en lo que respecta a las arquitecturas. Con las constantes actualizaciones, todos los demás parámetros también crecieron.

6. Conclusiones y Trabajos futuros

Con el desarrollo e inclusión de *eUCCvm* en diferentes cátedras de carreras de la UTN regional Mendoza a partir de 2018, la herramienta ha mostrado resultados satisfactorios tanto para la modalidad presencial como en la virtualidad. Los resultados fueron más allá del ámbito académico y la resolución de prácticos, también permite su aplicación en ambientes laborales y de investigación, resaltando en todos ellos su facilidad de uso y cantidad de arquitecturas soportadas. Es bueno destacar el rol fundamental adquirido durante el año 2020, con su utilización para la modalidad virtual.

Resulta importante resaltar que con *eUCCvm*, los estudiantes cuentan con una herramienta *"lista para utilizar"*, permitiendo optimizar el tiempo de clase al realizar actividades sin necesidad de descargar o instalación de programas adicionales. Además, la constante evolución de la plataforma, ya sea con el incremento año a año de la inclusión de nuevas arquitecturas y de actividades implementadas; o por su propiedad de ser software abierto y la inclusión de alguna mejora.

Como línea de trabajo futuro, existen varios aspectos a considerar, uno de ellos son llevar a cabo diferentes mejoras, entre ellas, se destaca la actualización de la máquina virtual a la última versión LTS de Ubuntu, así como la integración nativa de soporte para sistemas de 64 bits. Además, se planea incorporar la posibilidad de realizar actividades directamente en la plataforma mediante su integración web, lo que ofrecerá una experiencia más fluida y accesible para los usuarios.

Agradecimientos

Agradecemos el soporte de grupo GridTICs perteneciente a UTN - FRM y a los docentes de las cátedras Técnicas Digitales II y III por incluir *eUCCvm* o colaborar en su desarrollo. El resto del hardware necesario para el desarrollo de la plataforma ha sido obtenido por financiamiento particular de los autores del presente trabajo.

Referencias

1. BCM2835 chipset, Raspberry Pi documentation, http://bit.ly/2woaRKW 2. BCM2836 chipset, Raspberry Pi documentation, http://bit.ly/2HDmLH9
2. BCM2837 chipset, Raspberry Pi documentation, http://bit.ly/2K76gET

3. EDU-CIAA-NXP (Marzo 2017), http://www.proyecto-ciaa.com.ar/devwiki/doku.php?id=desarrollo:edu-ciaa:edu-ciaa-nxp
4. Blem, E., Menon, J., Sankaralingam, K.: A detailed analysis of contemporary ARMand x86 architectures. UW-Madison Technical Report (2013)
5. Cabo, J.M., Moralejo, R.O.: Desarrollo de instrumentos de evaluación educativahacia tecnologías específicas desde la perspectiva ciencia, tecnología y sociedad: El caso de la calidad del software. In: II Congreso de Tecnología en Educación y Educación en Tecnología (2007)
6. Coller, B.D., Scott, M.J.: Effectiveness of using a video game to teach a course inmechanical engineering. Computers & Education **53**(3), 900–912 (2009)
7. Dell'Oso, M., Lanzarini, L.C., Ridolfi, P.: Prototipo funcional de un sistema dedetección de caídas basado en la plataforma CIAA. In: XXII Congreso Argentino de Ciencias de la Computación (CACIC 2016). (2016)
8. D'Ausilio, A.: Arduino: A low-cost multipurpose lab equipment. Behavior researchmethods **44**(2), 305–313 (2012)
9. Goldberg, R.P.: Survey of virtual machine research. Computer **7**(6), 34–45 (June 1974). https://doi.org/10.1109/MC.1974.6323581
10. Griffiths, M.D.: The educational benefits of videogames. Education and health**20**(3), 47–51 (2002)
11. Hannig, F., Teich, J.: Open source hardware. Computer **54**(10), 111–115 (2021). https://doi.org/10.1109/MC.2021.3099046
12. Hyde, R.: The art of assembly language. No Starch Press (2003)
13. Ivkovi´c, J., Radulovi´c, B.: The advantages of using Raspberry Pi 3 compared toRaspberry Pi 2. SoC computers for sensor system support. In: International Conference on Applied Internet and Information Technologies. pp. 88–94 (2016)
14. Kuo, Y.M., Garcia-Herrero, F., de la Cuerda, J.A.M.: Design, implementation, andcharacterization of custom risc-v soft-core processors for future communication networks
15. Leens, F.: An introduction to I2C and SPI protocols. IEEE Instrumentation &Measurement Magazine **12**(1), 8–13 (2009)
16. Li, P.: Selecting and using virtualization solutions: our experiences with VMwareand VirtualBox. Journal of Computing Sciences in Colleges **25**(3), 11–17 (2010)
17. Meriac, M.: Heart of darkness-exploring the uncharted backwaters of hid iclass (tm) security. In: 27th Chaos Communication Congress (2010)
18. Moralejo, R.O., Cabo, J.M.: Visión de la tecnología en estudiantes de ingeniería ensistemas de información en mendoza-argentina. In: IX Workshop de Investigadores en Ciencias de la Computación (2007)
19. Pacey, A.: La cultura de la tecnología, méxico, fce. Enfoques sobre la Tecnologíaen: http://www. cneq. unam. mx/[Consulta: 21 de octubre de 2014] (1990)
20. Patterson, D., Waterman, A.: The RISC-V Reader: an open architecture Atlas.Strawberry Canyon (2017)
21. Perez-Monte, C.F., Diedrichs, A.L.: eUCCvm: Plataforma educativa de programación y compilacioón cruzada universal de sistemas embebidos (May 2019). https://doi.org/10.6084/m9.figshare.8185169.v1, https://doi.org/10.6084/m9.figshare.8185169.v1
22. Perez-Monte, C.F., Diedrichs, A.L.: eUCCvm: Plataforma educativa de programación y compilación cruzada universal de sistemas embebidos (7 2023). https://doi.org/10.6084/m9.figshare.8185169.v10, https://doi.org/10.6084/m9.figshare.8185169
23. Perez Monte, C.F., Diedrichs, A.L.: Programatecnicasdigitalesii-2023.pdf (Aug2023). https://doi.org/10.6084/m9.figshare.23895969, https://figshare.com/ articles/preprint/ProgramaTecnicasDigitalesII-2023_pdf/23895969/1

24. Perez Monte, C.F., Diedrichs, A.L.: Programatecnicasdigitalesiii-2023.pdf (Aug2023). https://doi.org/10.6084/m9.figshare.23895963, https://figshare.com/ articles/book/ProgramaTecnicasDigitalesIII-2023_pdf/23895963/1
25. Perez-Monte, C.F., Mercado, G., Taffernaberry, C., Diedrichs, A.L., Tobar, M.S.,Crespillo, E., Moralejo, R., Caballero, G., Gonzalez, R.: euccvm: Plataforma educativa de programación y compilación cruzada universal de sistemas embebidos. CASE 2019 (2019)
26. Prachchhak, G., Bhatt, C., Thik, J.: Data logging and visualization using bolt IoT.In: International Conference on Advanced Computing Networking and Informatics. pp. 155–164. Springer (2019)
27. Ritchie, D.M.: The development of the C language. ACM Sigplan Notices **28**(3), 201–208 (1993)
28. Sanchez, J., Canton, M.P.: Microcontroller programming: the microchip PIC. CRCpress (2006)
29. Silvia Lago Martínez, Ana Marotias, S.A.: Inclusión digital en la educación república argentina. el programa conectar igualdad. Revista Educación y Pedagogía **24**(62), 205–218 (2013), http://bit.ly/2wm5Ke3
30. Soper, M.E.: Raspberry Pi system anatomy. In: Expanding Your Raspberry Pi,pp. 1–15. Springer (2017)
31. Yiu, J.: The Definitive Guide to ARM® Cortex®-M3 and Cortex®-M4 Processors. Newnes (2013)

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Metodologías de evaluación de la participación en foros de debate académicos: Una revisión sistemática

Paula Dieser[1] [0000-0002-2066-0183], Cecilia Sanz[2,3] [0000-0002-9471-0008], Alejandra Zangara[2] [0000-0001-6013-9790]

Departamento de Matemática, Facultad de Ciencias Exactas y Naturales (UNLPam)
Instituto de Investigación en Informática LIDI (III-LIDI), Facultad de Informática (UNLP)
Comisión de Investigaciones Científicas de la Provincia de Buenos Aires
pauladieser@exactas.unlpam.edu.ar, csanz@lidi.info.unlp.edu.ar,
alejandra.zangara@gmail.com

Abstract. El foro de debate se emplea habitualmente para promover la interacción en propuestas de educación a distancia y mediadas por tecnología digital. Las tareas que allí se desarrollan podrían involucrar cognitivamente a los estudiantes durante su aprendizaje. Evaluar la forma en que ellos participan en estos espacios requiere de protocolos que permitan capturar características diversas de esta interacción. Este trabajo ofrece una revisión sistemática de la literatura que aborda estos temas en el ámbito de la Educación Superior. Con base en 54 investigaciones desarrolladas entre 2015 y 2020 se identifican cuatro metodologías (estadísticas simples, análisis de contenido, de redes sociales, y de procesos) que abordan diferentes aspectos (cantidad, calidad, temporalidad, y relaciones) de la participación. Se analizan y discuten los resultados con especial atención en las posibilidades que ofrecen estas metodologías para promover el aprendizaje constructivo en contextos de colaboración.

Keywords: foros, participación, metodologías de evaluación, estadísticas simples, análisis de contenido, análisis de redes sociales, análisis de procesos.

1. Introducción

La interacción es un elemento definitorio del hacer educativo e implica un proceso de comunicación [1]. En educación a distancia y, por extensión en cualquier relación educativa mediada, su abordaje ha sido, y es, objeto de constantes investigaciones [1].

Una herramienta habitual para promover la interacción académica y el aprendizaje en las propuestas educativas mediadas es el foro de debate [2, 3]. Éste es reconocido como un espacio de interacción con posibilidades para la promoción del pensamiento crítico, la resolución de problemas y la construcción de conocimiento [3, 4].

Los usos potenciales de estos espacios han generado múltiples investigaciones en torno a la temática. Parte de ellas se han centrado en la aplicación y desarrollo de metodologías que permiten analizar aspectos como la participación e interacción de quienes utilizan estos foros [2, 5] y su relación o impacto sobre otras variables como el rendimiento académico [2]. El estudio de estos temas permite identificar variables a considerar en el diseño, implementación y andamiaje de este tipo de tareas que favorezcan un buen desempeño del estudiante. Y es que, aunque el sostenimiento de un foro requiere de una participación activa [6], ésta no necesariamente implica un aprendizaje más profundo y colaborativo, ni una mejor calidad de los aportes [5].

Este trabajo revisa las metodologías empleadas en trabajos publicados o aceptados para su publicación entre 2015 y 2020 que estudian estos procesos en propuestas mediadas por tecnología digital en Instituciones de Educación Superior (IES). En particular, se indaga ¿qué protocolos se han empleado para evaluar la participación en actividades de foro, y qué aspectos o dimensiones éstos evalúan?

Como instancia previa a esta revisión, se presentan algunos elementos teóricos que la sustentan. En particular, aspectos del microcontexto tecnológico de interés, *i.e.* un foro de debate, y el constructo a evaluar en este espacio. Luego, se expone la metodología de revisión empleada, *i.e.* detalles del proceso de búsqueda, y los criterios de inclusión y exclusión adoptados. Posteriormente, se describen los artículos que conforman el *corpus* de revisión, para continuar con el análisis de los resultados alcanzados en torno a las metodologías empleadas para evaluar diversos aspectos de la participación. Finalmente, se presentan algunas conclusiones y se reseñan las líneas de trabajo futuro, en el contexto de una tesis de maestría, derivadas de los resultados alcanzados en esta revisión.

2. Participación en foros de debate académicos

Un foro se define como un espacio virtual comunicativo y colaborativo en el que un grupo toma parte en un debate sobre un tema de interés común [7]. Se entiende como un lugar de expresión, opinión y consulta de sus integrantes [7] que propicia el intercambio, la concertación y el consenso de ideas [8].

Se trata de una herramienta de comunicación asincrónica basada en texto escrito [7, 8] que tiene lugar en un espacio *web* y puede estar moderada [3, 7]. Su condición mínima de apertura y sostenimiento es la participación, que puede restringirse a algunos integrantes de la comunidad (foro privado) o no (foro público). A su vez, puede o no ser anónima, según se identifique al autor de cada aporte [7].

En educación, los foros han cobrado auge como herramienta de participación y comunicación en los entornos virtuales de enseñanza y aprendizaje (EVEA). Desde allí, cada participante accede al registro de los aportes ya realizados y puede generar otros nuevos. En consecuencia, no es posible identificar la participación sólo con los mensajes que se emiten en este espacio. Según [6] la participación es “el proceso por el que las intervenciones se relacionan y producen un sentido que sobrepasa la simple sucesión de mensajes” (p. 3). Este proceso se genera al ponerse en relación los mensajes, los textos, los emoticones, el contexto, la audiencia, entre otros elementos.

Quienes forman parte de un foro participan a partir de intervenciones que se relacionan y producen un sentido (participación activa) o mediante el acceso para mirar la lista de mensajes publicados (participación pasiva) [6]. En el mismo sentido, [9] habla de una interacción vicaria. Se refiere a ella en términos de un estudiante que procesa activamente las interacciones de otros sin dejar rastros visibles en la herramienta mediadora. Esta forma de interacción, en contraposición a la directa, junto con las categorías actor y no actor, ofrece una clasificación de los estudiantes en condiciones de interactuar en procesos educativos mediados [9]. Un estudiante es actor si hace aportes unilaterales sin considerar las reacciones o comentarios de otros; y es no actor si no participa de modo alguno [9]. Estas ideas se resumen en la Fig. 1.

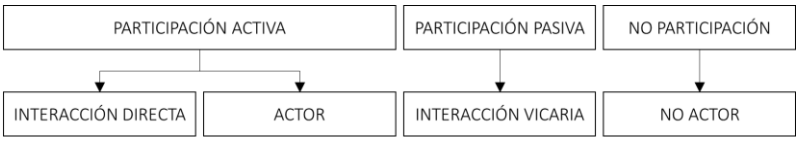


Fig. 1. Correspondencia entre tipos de participación [6, 9]. Elaboración propia.

El debate mediado por foros tiene el potencial de involucrar cognitivamente a los estudiantes durante su aprendizaje [4]. Lo ubican en el centro del proceso formativo y se orientan al desarrollo del pensamiento crítico, la resolución de problemas y la construcción conjunta de conocimiento [3]. Al mismo tiempo, pueden emplearse para promover habilidades comunicativas, sociales y de colaboración, la metacognición, la gestión del tiempo y la autorregulación del aprendizaje [10]. Pese a ello, es frecuente la falta de participación de los estudiantes en estos espacios [11]. En consecuencia, resulta de interés estudiar metodologías orientadas a identificar patrones de participación que pudieran derivar en mejores resultados y desempeños estudiantiles.

3. Metodología

A fin de atender el objetivo de este trabajo, se realizó una revisión sistemática [12] de estudios empíricos, siguiendo los lineamientos de [13] renovados según los aportes de [14]. A continuación, se describen la estrategia de búsqueda para la recuperación de los estudios iniciales y los criterios de selección de los estudios incluidos en el *corpus*.

3.1 Estrategia de búsqueda

Se realizó una búsqueda automática, mediante las herramientas de búsqueda avanzada de EBSCOhost sobre las siguientes bases de datos, bibliotecas digitales, y portales de revistas: *Academic Search Premier*, *Scopus*, *Education Resources Information Center* (ERIC), *Education Full Text*, *Directory of Open Access Journals* (DOAJ), *IEEE Xplore Digital Library*, *Scientific Electronic Library Online* (SciELO), *ScienceDirect*, JSTOR *Journals*, y editoriales incluidas en *Complementary Index* (Elsevier, Wiley, Springer, Taylor & Francis, Sage, ACM Digital Library). También se emplearon técnicas de *snowballing* [15] e incorporaron estudios sugeridos por expertos o identificados mediante una búsqueda manual sobre las tablas de contenido de las revistas *American Journal of Distance Education*, *Journal of New Approaches in Educational Research* (NAER), Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología (TE&ET), y actas de jornadas y congresos con ejes temáticos vinculados con el uso pedagógico de tecnologías digitales disponibles en el Repositorio Institucional de la Universidad Nacional de La Plata (SEDICI).

La cadena de búsqueda resultó una expresión booleana de los términos listados en la Tabla 1 (*OR* A*) *AND* (*OR* B*) *AND* (*OR* C*). Sobre los trabajos identificados se aplicaron filtros a fin de restringirlos al contexto de las IES dados por *higher education*, *postsecondary education*, *undergraduates*, y *undergraduate student*.

Tabla 1. Términos de búsqueda según temas y contexto de interés. Elaboración propia.

FORO DE DEBATE		EVALUACIÓN		EDUCACIÓN MEDIADA	
A1	*discussion forum*	B1	*assessment*	C1	*digital learning environment*
A2	*discussion board online*	B2	*evaluation*	C2	*e-learning*
A3	*discussion*	B3	*achievement*	C3	*online learning*
A4	*asynchronous communications*	B4	*quantitative method*	C4	*virtual learning*
A5	*asynchronous discussion*	B5	*qualitative method*	C5	*technology enhanced learning environment*
A6	*computer mediated communication virtual*	B6	*descriptive statistics*	C6	*web-based learning environment*
A7	*community of inquiry*	B7	*sequential analysis*	C7	*learning management system*
		B8	*content analysis*	C8	*ICT mediated environment*
		B9	*social network analysis*	C9	*ICT mediated learning*
				C10	*ICT mediated education*

3.2 Criterios de selección

Se tomaron publicaciones de texto completo, arbitradas, en inglés o español, publicadas o aceptadas para su publicación entre enero de 2015 y diciembre de 2020, que consideren estudiantes de IES como población de interés. En consecuencia, los criterios de exclusión de trabajos quedaron determinados por su carácter no arbitrado, por su idioma, por no corresponder a la temática abordada, o por no referirse a IES.

4. Resultados y discusión

Se describen a continuación las principales características de los artículos incluidos en el *corpus* y los resultados que responden las preguntas de investigación formuladas.

4.1 Descripción de artículos seleccionados

Como se esquematiza en la Fig. 2, la estrategia de búsqueda permitió recuperar 401 documentos. Por contraste del identificador de objeto digital (DOI) y la referencia bibliográfica se identificaron 53 duplicaciones. Sobre los 348 artículos resultantes se aplicaron los criterios de inclusión a partir de la lectura del título y resumen, descartándose 259 trabajos. Luego, se realizó la lectura completa de los 89 artículos potencialmente elegibles. Este procedimiento generó el descarte de 35 trabajos por no atender de forma completa a los criterios de inclusión definidos, y la selección final de 54 artículos para su análisis y síntesis. La tabla accesible mediante el código QR de la Fig. 2 incluye sus referencias y un resumen de sus principales características.

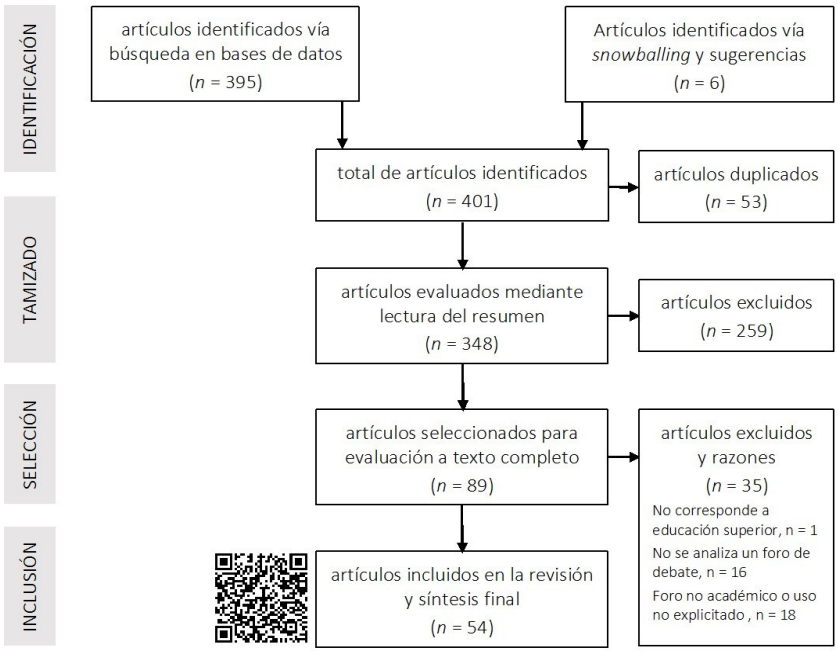


Fig. 2. Diagrama de búsqueda, identificación y selección de artículos. Características de los 54 artículos del *corpus* de revisión https://acortar.link/XPu53C. Elaboración propia.

4.2 Metodologías de evaluación de la participación

La revisión bibliográfica ha permitido identificar cuatro metodologías (estadísticas simples, análisis de contenido, de redes sociales, de procesos) que evalúan diversos aspectos (cantidad, calidad, temporalidad, relaciones) de la participación (Fig. 3). Gran parte de las investigaciones (n = 24; 44,4%) combinan protocolos. Los trabajos restantes (n = 30, 55,6%) emplean una única forma de evaluación. En los parágrafos siguientes se describen las cuatro metodologías identificadas.

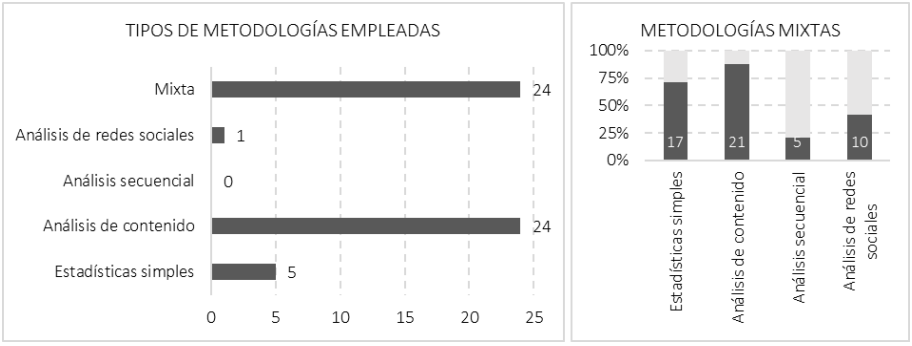


Fig. 3. Tipos de metodologías empleadas en 54 investigaciones del *corpus*. Elaboración propia.

Análisis mediante estadísticas simples. Los sistemas *web* que albergan los foros posibilitan la trazabilidad de la interacción de un usuario con la herramienta. El análisis de estos datos es un medio eficiente de la minería de datos y la analítica del aprendizaje para comprender la participación [16]. A partir de estos datos, es posible generar estadísticas que, entre otras funcionalidades, permiten obtener un sistema de alerta o derivar intervenciones pedagógicas para que estudiantes y docentes puedan notar el progreso del aprendizaje y predecir resultados en forma temprana [16].

Un total de 22 trabajos del *corpus* (40,7%) emplean esta metodología (Fig. 3) para evaluar la cantidad, la temporalidad y la calidad de la participación mediante diversos indicadores descriptos a continuación. La mayoría de estas investigaciones (n = 17; 77,2%) usa las estadísticas simples en conjunto con alguna otra forma de evaluación.

1. *Indicadores de cantidad de participación*. Los más usados se obtienen sobre la frecuencia de acceso al foro, de lectura y publicación de mensajes. El número total de visitas al foro, sin distinguir el tipo de actividad (lectura o escritura), indica la intención de participación [16]. También suele emplearse una media de visitas en un periodo dado (diario o semanal). Cuando se analiza el tipo de acción realizada, el número total de publicaciones es un indicador popular. Determina el interés en un tema particular y es indicativo de la participación activa en el debate [16]. En ocasiones se generan estadísticas según el tipo de publicación, como el número de mensajes que inician hilos o de respuestas a estos aportes iniciales. Para considerar la participación pasiva, se usa el total de aportes vistos. Los estudiantes pueden ser grandes oyentes, ingresan al foro y ven las publicaciones aun cuando no realicen aportes en él [17].
2. *Indicadores temporales de participación*. Complementan los indicadores cuantitativos. Se obtienen a partir de las marcas de tiempo (fecha y hora) de los eventos considerados. Una métrica simple es el tiempo total pasado en el foro para leer las publicaciones de otros o para escribir las propias. Al usarse en estudiantes que no hacen aportes es una forma de valorar su participación pasiva [17]. La media de los intervalos de acceso al foro ofrece una métrica de regularidad en relación con el aprendizaje autorregulado [17]. Por último, es posible examinar la puntualidad por función o rol mediante un índice binario según se haya cumplido o no con la tarea asignada en los tiempos pautados.
3. *Indicadores de calidad de participación*. Valoran la participación activa y de calidad. La cantidad media de vistas y de colaboradores en los hilos iniciados por un estudiante, y el número total de respuestas a un mensaje son formas de recompensar a quienes iniciaron hilos importantes y que, además, involucraron a muchos miembros [17]. Los votos y respaldos articulan juicios humanos y evitan que el foro se inunde con hilos sin sentido a fin de aumentar el recuento de aportes [17]. A partir de estos registros, se considera el número de mensajes del estudiante valorados o respaldados. Otras métricas son la profundidad del mensaje (nivel del mensaje dentro de un hilo) o del hilo (nivel máximo de respuesta en el hilo). Éstas permiten obtener patrones de crecimiento de los hilos basados en una taxonomía (corto, extendido, dividido) [11].

Análisis de contenido. La transcripción de los aportes de los estudiantes constituye un material con riqueza para estudiar los mecanismos de aprendizaje [18] en cada una

de sus dimensiones cognitiva, metacognitiva, afectiva y motivacional [19]. Por ello, múltiples autores han usado este protocolo para evaluar la calidad de la participación.

En efecto, 45 son los trabajos del *corpus* que usan la metodología (Fig. 3). Casi la mitad (n = 21; 46,7%) la emplean en combinación con otra forma de evaluación a fin de analizar la dinámica de la participación de los estudiantes y sus interacciones [16].

Todos estos trabajos avanzan sobre la dimensión cuantitativa de la metodología,

i.e. calculan frecuencias de los fenómenos de interés mediante un análisis cualitativo de las transcripciones. Esto permite registrar y categorizar el contenido de la comunicación, y comparar y contrastar los resultados de la codificación [16]. La dimensión cualitativa se caracteriza por el método (manual o automático) y el enfoque de codificación (*a priori* o emergente) a aplicar sobre la unidad de análisis elegida (mensaje, unidad temática o unidad sintáctica). A continuación, se reseñan los esquemas manuales y automáticos empleados en las investigaciones del *corpus*.

1. *Esquemas manuales*. Los usados más frecuentemente son de amplia difusión y suelen combinarse entre sí. Estos esquemas evidencian comportamientos similares relacionados con la profundidad e intensidad de la cognición [20]. En particular, se analiza el aprendizaje desde un enfoque cognitivo mediante el método de [18], evidencias de pensamiento crítico a través del esquema de [21], y de construcción de conocimiento por medio del Modelo de Análisis de Interacción de [22]. Con base en el Modelo de Comunidad de Indagación [23] se desarrollan esquemas para evaluar las presencias cognitiva, social y docente. A fin de revelar procesos cognitivos se emplea la taxonomía revisada de Bloom [24]. Otros esquemas se usan en un único artículo para interpretar actos de habla, evaluar la calidad de los mensajes, entre otros aspectos.
2. *Esquemas automáticos*. La codificación manual no puede gestionar con éxito el gran volumen de datos generados en los foros. Esto requiere procedimientos de codificación automáticos. Sólo cinco trabajos del *corpus* (11,1%) han desarrollado estas formas automáticas y muestran el potencial de las técnicas de minería de datos y del procesamiento del lenguaje natural para evaluar diferentes aspectos del proceso de aprendizaje en estos espacios. En particular, construcción de conocimiento, presencia cognitiva, sentimientos, actos de habla y temas destacados. Uno de los desafíos de la automatización del análisis de contenido es que los constructos importantes en educación son latentes y no están explícitamente presentes en las transcripciones [19]. No obstante, abren las puertas al desarrollo de sistemas de retroalimentación automatizados en tiempo real sobre áreas que requieren intervención [19].

Análisis secuencial de procesos. Examina la significancia de la ocurrencia secuencial de conductas y permite visualizarlas mediante diagramas de transición [25]. En particular, porque los procesos de interacción en los foros son acumulativos y la secuencia en la que se encuentran las conductas afecta cómo y qué se aprende [26].

Estas conductas suelen identificarse mediante análisis de contenido de los aportes en los foros. Por ello, ambas metodologías se usan en forma conjunta en cinco (11,1%) trabajos del *corpus* (Fig. 3). En éstos se ha empleado el análisis secuencial para identificar secuencias de construcción de conocimiento, de procesamiento cognitivo, y de actos de habla.

Análisis de redes sociales. Examina los atributos estructurales de las relaciones que se establecen [27] y se utiliza para identificar a los individuos centrales, prominentes o prestigiosos, dentro de la red que representa el foro. Estos atributos reflejan, además, aspectos de la cohesión grupal y de la cantidad y calidad de las interacciones.

Sin embargo, esta metodología ha recibido poca atención de los investigadores [28] y, por sí sola, no es suficiente para comprender en profundidad los patrones de interacción en un entorno de aprendizaje dado [19]. Para lograr esta comprensión integral, debe aplicarse como un enfoque complementario al análisis de contenido y acompañado de otros indicadores como estadísticas simples [19]. Tal es el caso de 10 de 11 trabajos del *corpus* de revisión (20,4%) que usan el análisis de redes sociales para evaluar diversos aspectos de las redes generadas en los foros educativos (Fig. 3).

En estas investigaciones se ha empleado una diáspora de métricas para evaluar múltiples aspectos con base en las relaciones direccionales y no direccionales.

1. *Centralidad de actor y de red.* Las métricas usadas con mayor frecuencia son el grado, la cercanía, y la intermediación.
2. *Cohesión grupal*. Se caracteriza mediante la densidad y la centralidad de red.

Otros instrumentos de utilidad son el sociograma y las representaciones matriciales [27]. Ambos permiten comprender la construcción social del aprendizaje, la dinámica de la clase, y facilitan la identificación de estudiantes en riesgo. En consecuencia, son herramientas que respaldan la planificación y la toma de decisiones [16, 28].

5. Conclusiones y trabajo futuro

Los foros de debate constituyen un espacio virtual facilitador de la interacción con posibilidades para promover el pensamiento crítico, la construcción de conocimiento y la resolución de problemas. Estos usos potenciales requieren de la participación de los estudiantes. Estudiar la forma en que ellos interactúan en estos espacios es un tema de constante interés en el ámbito académico y científico.

En este trabajo se indagan formas de evaluar la participación en foros en 54 investigaciones desarrolladas en IES y publicadas entre 2015 y 2020. Estas iniciativas dan cuenta de cuatro metodologías, de uso no excluyente, que permiten evaluar diferentes aspectos del constructo. Trabajos posteriores [29, 30] avanzan en la misma línea y demuestran perspectivas similares.

Entre ellas, el análisis manual de contenido para evaluar la calidad de los aportes, es el método más popular. Debido a las dificultades y el tiempo que consume este procesamiento, se han desarrollado algunos protocolos automáticos que dan cuenta de las posibilidades del aprendizaje automático y del procesamiento de lenguaje natural para evaluar el proceso de aprendizaje y la colaboración en los foros de debate.

El análisis de contenido suele acompañarse del análisis secuencial de las intervenciones para dar cuenta de la temporalidad y el orden de estas últimas. Este último se usa con menor frecuencia entre las investigaciones del *corpus* de revisión.

Como complemento de estas dos metodologías, el cálculo de estadísticas simples a partir de los datos de registro es reconocido como eficiente en el campo de la minería de datos educativos y la analítica del aprendizaje. En particular, por la facilidad de su

obtención e interpretación. De allí que sea la segunda metodología más usada entre los trabajos revisados que, además, permite dar cuenta de la cantidad de participación.

Por último, el análisis de redes sociales posibilita detectar patrones y visualizar la interacción de los miembros. Esto permite a los profesores percibir aspectos implícitos de los datos brutos y apoyar su conciencia sobre el proceso de colaboración subyacente. Así, se promueven intervenciones educativas oportunas y flexibles.

Al considerar los beneficios y las limitaciones de cada metodología, la integración de diversos protocolos es una tendencia de investigación que puede contribuir a una mejor comprensión de la participación de los estudiantes en los foros de debate.

Los resultados de esta revisión junto a los de una revisión previa [31] derivan en el desarrollo de una metodología integral que indaga acciones de participación de un grupo de estudiantes de postgrado en una tarea de foro, estrategias de autorregulación del aprendizaje [32] y sus relaciones con el rendimiento académico. Esta metodología constituye uno de los principales aportes de una tesis de maestría.

Referencias

1. García Aretio, L.: Fundamento y componentes de la educación a distancia. Revista Iberoamericana de Educación a Distancia, 2(2), 28-39 (1999).
2. Almatrafi, O., Johri, A.: Systematic review of discussion forums in Massive Open Online Courses (MOOCs). IEEE Transactions on Learning Technologies, 12(3), 413-428 (2019).
3. Roig, R., Rosales, S.: Assessing participation in online discussion forums: A proposal for multidimensional analysis. Píxel-Bit, 40, 137-149 (2012).
4. Wang, Y.: Enhancing the quality of online discussion: Assessment matters. Journal of Educational Technology Systems, 48(1), 112-129 (2019).
5. Gros Salvat, B., Silva, J.: El problema del análisis de las discusiones asincrónicas en el aprendizaje colaborativo mediado. Revista de Educación a Distancia, 16 (2006).
6. Núñez Mosteo, F., Gálvez Mozo, A., Vayreda, A.: La participación en un foro electrónico: motivos, auditorios y posicionamientos. REDcientífica, 47, 1-14 (2003).
7. Sanz, C., Zangara, A.: Los foros como espacios comunicacionales - didácticos en un curso a distancia. Una propuesta metodológica para aprovechar sus potencialidades. En: Actas XII CACIC, Potrero de los Funes, San Luis, pp. 1021-1033 (2006).
8. Arango, M.: Foros virtuales como estrategia de aprendizaje. Debates latinoamericanos, 2, 85-105 (2004).
9. Sutton, L.: The principles of vicarious interaction in computer-mediated communication. Journal of Interactive Educational Communications, 7(3), 223-242 (2001).
10. Gašević, D., Adesope, O., Joksimović, S., Kovanović, V.: Externally facilitated regulation scaffolding and role assignment to develop cognitive presence in asynchronous online discussions. Internet and Higher Education, 24, 53-65 (2015).
11. Ghadirian, H., Ayub, A., Bakar, K., Hassanzadeh, M.: Growth patterns and e-moderating supports in asynchronous online discussions in an undergraduate blended course. International Review of Research in Open and Distributed Learning, 17(3), 189-208 (2016).
12. Sánchez Meca, J.: Cómo realizar una revisión sistemática y un meta-análisis. Aula Abierta, 38(2), 53-63 (2010).
13. Petticrew, M., Roberts, H.: Systematic reviews in the Social Sciences: A practical guide. Blackwell Publishing, Oxford (2006).
14. Lavallée, M., Robillard, P. N., Mirsalari, R.: Performing systematic literature reviews with novices: An iterative approach. IEEE Transactions on Education, 57(3), 175-181 (2014).

15. Greenhalgh, T., Peacock, R.: Effectiveness and efficiency of search methods in systematic reviews of complex evidence: Audit of primary sources. British Medical Journal, 331(7524), 1064-1065 (2005).
16. Jo, I., Park, Y., Lee, H.: Three interaction patterns on asynchronous online discussion behaviours: A methodological comparison. Journal of Computer Assisted Learning, 33(2), 106-122 (2017).
17. Bihani, A., Paepcke, A.: QuanTyler: Apportioning credit for student forum participation. En International Conference on Educational Data Mining, Raleigh (2018).
18. Henri, F.: Computer conferencing and content analysis. En Kaye, A. (ed.) Collaborative learning through computer conferencing, pp. 117-136. Springer-Verlag, London (1992).
19. Joksimović, S., Jovanović, J., Kovanović, V., Gašević, D., Milikić, N., Zouaq, A., Van Staalduinen, J.: Comprehensive analysis of discussion forum participation: from speech acts to discussion dynamics and course outcomes. IEEE Transactions on Learning Technologies, 13(1), 38-51 (2019).
20. O'Riordan, T., Millard, D. E., Schulz, J.: How should we measure online learning activity? Research in Learning Technology, 24 (2016).
21. Newman, D. R., Webb, B., Cochrane, C.: A content analysis method to measure critical thinking in face-to-face and computer supported group learning. Interpersonal Computing and Technology, 3(2), 56-77 (1995).
22. Gunawardena, C., Lowe, C., Anderson, T.: Analysis of a global online debate and the development of an interaction analysis model for examining social construction of knowledge in computer conferencing. Journal of Technical Writing and Communication, 17(4), 397-431 (1997).
23. Garrison, D., Anderson, T., Archer, W.: Critical inquiry in a text-based environment: Computer conferencing in Higher Education. The Internet and Higher Education, 2(2-3), 87-105 (2000).
24. Anderson, L., Krathwohl, D.: A taxonomy for learning, teaching, and assessing. A revision of Bloom's taxonomy of educational objectives. Addison Wesley, New York (2001).
25. Wu, S. Y., Hou, H. T.: How cognitive styles affect the learning behaviors of online problem-solving based discussion activity: A lag sequential analysis. Journal of Educational Computing Research, 52(2), 277-298 (2015).
26. Reimann, P.: Time is precious: variable -and event- centred approaches to process analysis in CSCL research. Computer Supported Collaborative Learning, 4, 239–257 (2009).
27. Wasserman, S., Faust, K.: El análisis de las redes sociales en las ciencias sociales y del comportamiento. En Análisis de redes sociales: Métodos y Aplicaciones, pp. 35-58. Centro de Investigaciones Sociales, Madrid (2013).
28. Ghadirian, H., Salehi, K., Ayub, A.: Analyzing the social networks of high and low performing students in online discussion forums. American Journal of Distance Education, 32(1), 27-42 (2018).
29. Liu, Z., Zhang, N., Peng, X., Liu, S., Yang, Z.: Students' Social-Cognitive Engagement in Online Discussions: An Integrated Analysis Perspective. Educational Technology & Society, 26(1), 1-15 (2023).
30. Norz, L., Dornauer, V., Hackl, W., Ammenwerth, E.: Measuring social presence in online based learning: An exploratory path analysis using log data and social network analysis. The Internet and Higher Education, 56 (2023).
31. Dieser, P. Sanz, C., Zangara, A.: Metodologías para la evaluación de la autorregulación del aprendizaje en contextos educativos mediados por tecnología digital. Una revisión sistemática. En: Actas XVII Congreso TE&ET, pp. 26-35 (2022).
32. Dieser, P. Sanz, C., Zangara, A.: Hacia una comprensión dinámica y contextual de la autorregulación del aprendizaje: Propuesta metodológica para su evaluación en foros académicos. En: Actas XVIII Congreso TE&ET, pp. 42-51 (2023).

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Generación de STEAM en condiciones extremas con xSTEAM

Gilda R. Romero[1, 3, 4], Cecilia Salina[1], Martín Domen[1], Daniela López De Luise[1, 2, 3], Mauro Bejarano[1,5]

[1] Centro de Altos Estudios en Tecnología Informática. Facultad de Tecnología Informática. Buenos Aires, Argentina.

[2] CI2S Labs. Buenos Aires, Argentina

[3] Sociedad Científica Argentina. Buenos Aires, Argentina

[4] Universidad de la Cuenca del Plata, Facultad de Ingeniería y Tecnología

[5] UCSE DASS. San Salvador de Jujuy, Argentina

gilda.romero@gmail.com, cecilia.salina@alumnos.uai.edu.ar, martin.dome@alumnos.uai.edu.ar, mdldl@ci2s.com.ar, maurobejarano97@gmail.com

Abstract. Las experiencias STEAM en la educación son empleadas para promover nuevas vocaciones en el ámbito científico y tecnológico, se caracterizan por su innovación tanto en estrategias pedagógicas como en las prácticas grupales y suelen implicar usos tecnológicos avanzados sino específicos. Sin embargo, las investigaciones del proyecto LINCIEVIS en el sector muestran que en ocasiones existe la necesidad de articular dichas experiencias de maneras alternativas. Este trabajo presenta un diseño de ciclo de vida y la protocolización de estos casos especiales denominados xSTEAM. En él se identifican las principales variables y las dinámicas sobre los elementos de trabajo que deben considerarse para afrontar las carencias eventuales en contextos desfavorables; también se presenta y analiza un caso de estudio. La nueva modalidad permite incorporar experiencias STEAM en condiciones dramáticamente desventajosas en el plano tecnológico.

Keywords: STEM, STEAM, xSTEAM, tecnología, educación.

1. Introducción

En septiembre de 2015 la Organización de las Naciones Unidas (ONU) desarrolló la Agenda 2030 para un desarrollo sostenible. Dicha agenda cuenta con 17 objetivos (denominados ODS) [1], como una llamada a la acción a los países para erradicar la pobreza, proteger al planeta, garantizar la paz y la prosperidad. Los ODS son una evolución respecto a los Objetivos de Desarrollo del Milenio, que comenzaron en el año 2000 para abordar la temática de la pobreza mundial.

Diversos autores incorporan el uso del arte como vehículo en el proceso cognitivo y volitivo, aumentando la adaptabilidad, creatividad en las nuevas generaciones habilidades que deben ser desarrolladas según lo confirma UNESCO [2], para enfrentar la adquisición y generación de tecnologías por venir [3].

La educación STEAM refiere a una educación integradora de las materias que lo componen. Para Moore y Smith [4] combina las disciplinas ciencia, tecnología, ingeniería y matemática en una clase, unidad o lección que está basada en un conjunto de conexiones entre dichas disciplinas y los problemas del mundo real.

Entre las características observadas en la educación STEAM se encuentran:

- garantizar la transversalidad del proceso de enseñanza aprendizaje.
- mejorar la contextualización de la aplicación de ciertos principios.
- generar un aprendizaje significativo, pues adquiere relevancia e interés.
- desarrollar habilidades interpersonales, creatividad y resolución de problemas.

Las prácticas STEAM se caracterizan por una fuerte innovación tanto en las estrategias pedagógicas como en las prácticas grupales, exigiendo romper con varios paradigmas de la "educación tradicional". En este trabajo se presenta el estudio de las barreras y una estrategia de solución, en condiciones de aplicación con menor acceso a tecnologías informáticas. En las próximas secciones se presentan una serie de condicionantes, cuyo nivel de impacto se miden a través de ciertas variables. También se describe un protocolo de aplicación, control y monitoreo tanto del proceso STEAM durante su realización como del modo de convertirlo en una unidad de valor aplicable para otros interesados -distintos de los que intervienen-. Estas consideraciones se desarrollan y plantean un nuevo tipo de STEAM, al que los autores denominan xSTEAM (Extreme STEAM).

A fin de cumplir con lo descrito, se expone un caso de uso, su análisis y las consideraciones que deben ser tenidas en cuenta al desarrollar una propuesta xSTEAM, donde se deben afrontar las carencias eventuales en contextos desfavorables. El trabajo se ejecuta en el marco del laboratorio LINCIEVIS (Laboratorio de INvestigaciones CIEntíficas, VIdeo juegos y STEAM, auspiciado técnicamente por la Sociedad Científica Argentina), y del CI2S Labs (Computational Intelligence & Information Systems Labs), en colaboración con la Universidad Autónoma de Entre Ríos con sede en Concepción del Uruguay, el IEEE UCSE DASS y el Centro de Altos Estudios en Tecnología Informática (CAETI) de la Universidad Abierta Interamericana (UAI).

El trabajo se organiza como sigue: la sección 2 introduce conceptos sobre lo que es STEAM y sus competencias, la sección 3 describe brevemente la investigación global en la que se enmarca este trabajo, la sección 4 presenta el caso de uso, el proceso de derivación de variables, las restricciones para la implementación del protocolo y un análisis resumido de los hallazgos. La sección 5 expone conclusiones y trabajos futuros.

2. STEAM: Definiciones y Competencias STEAM

El modelo de aprendizaje o educación STEAM es una educación integradora de materias relacionadas con la ingeniería y ciencias, y a la vez pretende promover estas áreas. Como destacan López De Luise y Tabares en [5] estos modelos renuevan los procesos áulicos y potencian paradigmas científicos y tecnológicos. Vásquez Giraldo [7] detalla la necesidad de alfabetizar de una forma interdisciplinaria en todos los niveles de la educación, requiere de personas que dominarán y fueran capaz de aplicar varias disciplinas integradamente, siendo base del desarrollo económico de los países industrializados. La National Science Foundation (NSF) y el National Research Council (NRC) del gobierno de Canadá, que en 2011 declararon estas disciplinas como las fundamentales para las sociedades tecnológicamente avanzadas o en proceso de llegar a la tecnificación. Ellas destacan que STEM contribuye a conseguir una mayor competitividad y, por consiguiente, ayudan a conseguir una mayor prosperidad

económica en el futuro y es son un claro indicador de la capacidad de un país para sostener un crecimiento continuo [8]. En forma análoga varios países las promueven. Concretamente en Argentina existen diversas iniciativas que forman parte del Plan Nacional de Ciencia, Tecnología e Innovación 2030 (PNCTI) [9].

En cuanto a las competencias desarrolladas, Tsupros et. al. [10] lo prescriben para impartir conceptos académicos complejos, asociar a la vida diaria aplicaciones de la ciencia, la tecnología, la ingeniería y las matemáticas, conectar la escuela, la comunidad, el trabajo y la empresa global, y desarrollar la capacidad de competir en la nueva economía por medio de habilidades STEM.

Finalmente, a fines de la década del 2010, como una nueva evolución del modelo comienza a desarrollarse el concepto de STEAM+H, mencionados en los 17 ODS de la Agenda 2030 de la ONU. El nuevo acrónimo resulta de añadir la H de Humanidades, a la sigla ya existente. Para la UNESCO [11] este nuevo enfoque dota a la educación STEAM de una perspectiva que orienta los conocimientos de ciencias y objetivos en pos de la resolución de "retos globales", tales como el cambio climático y la pobreza. Los estudiantes tendrían que contar con capacidades (conocimientos, competencias, valores y actitudes) para construir un mundo más justo, pacífico, tolerante, inclusivo, sostenible y seguro. En este sentido, por ejemplo, cada estudiante debería ser competentes para aprovechar las tecnologías 4.0, favorecer a la productividad y a la prosperidad del lugar donde viven desde un compromiso con los derechos humanos, la paz, la justicia y el Estado de Derecho. Así la ONU entiende con ésto que es necesaria la educación STEAM+H para cultivar un pensamiento y habilidades transformadoras, innovadoras y creativas para avanzar hacia el desarrollo sostenible.

3. El proyecto de xSTEAM

El grupo LINCIEVIS de la Sociedad Científica Argentina estudia y aplica experiencias STEAM desde el año 2018, tanto en Argentina como en otros países. Las investigaciones desarrolladas han derivado en varias publicaciones identificado diferentes variables y elementos de trabajo. En ese recorrido han emergido situaciones especiales como las del caso de estudio que se describe en la próxima sección. Se presenta entonces un punto de inflexión que debe considerar cómo afrontar carencias eventuales severas en contextos desfavorables. A este nuevo tipo de experiencias STEAM se les ha dado en llamar xSTEAM (Extreme STEAM). Este trabajo expone un caso típico, su análisis y cómo la propuesta xSTEAM enfrenta y resuelve el caso.

La investigación del proyecto comprende para estos casos: las delimitaciones de situaciones xSTEAM, la determinación de hipótesis tecnológicas pertinentes, valoración de las variables específicas de control, y la formalización de un protocolo que adapta procesos STEAM a situaciones especiales de escasa o nula tecnología.

3.1 Determinación de un caso xSTEAM

A fin de que una experiencia STEAM sea considerada extrema (xSTEAM) deben cumplirse ciertas condiciones específicas que delimitan los casos y los encuadran adecuadamente. Los estudios realizados permiten identificar las condiciones:

- Los participantes que realizarán la experiencia STEAM se hallan en situación de baja accesibilidad tecnológica.
- La conectividad es dramáticamente deficiente.
- Existe una restricción cultural que condiciona la implementación de experiencias STEAM, por caso prejuicios respecto a la implementación tecnológica.
- Los docentes y/o facilitadores del proceso de enseñanza aprendizaje carecen de competencias para el desarrollo de este tipo de experiencias.
- Los mediadores tienen poca o nula información acerca de implementaciones STEAM y/o cómo pueden medir su impacto.
- Carencia de un protocolo sistemático que permita evaluar la eficacia del proceso y/o desarrollo de actividades STEAM.
- Desconocimiento de las variables endógenas y exógenas que hacen a la experiencia STEAM.

Es de notar que para encuadrarse en situación extrema pueden darse tres o más de las condiciones citadas en el listado anterior. Respecto a los dos últimos, un cuidadoso estudio bibliográfico permite afirmar que frecuentemente las instituciones aplican STEAM sin cumplir con estos requisitos; sin embargo, éstos son esenciales para poder evaluar y administrar adecuadamente el post-mortem de las actividades y estimar de manera precisa y sistemática los resultados [5].

3.2 Hipótesis tecnológicas

Una vez identificada la experiencia dentro del rango xSTEAM, deben considerarse las circunstancias tecnológicas específicas de modo tal que pueda abordarse con las herramientas y/o frameworks adecuados para soportar a estos casos extremos y alternativos a los tradicionales abordajes.

El proceso entonces continúa con la búsqueda de los FCE (Factores Críticos de Éxito) para la implementación del proyecto STEAM en específico. La descripción de este proceso se halla fuera del alcance de este trabajo, pero es importante notar que el foco al corto y/o mediano plazo es la mejora de los aprendizajes en las personas involucradas.

El listado FCE, busca identificar qué tecnologías especiales deben considerarse, y la eventual posibilidad de abordarse con frameworks que den soporte a estos casos extremos y alternativos a los tradicionales abordajes. Es importante destacar en este punto que la preferencia por el uso de frameworks se debe a que son quienes facilitan la replicabilidad y aceleran la implementación. En el caso de LINCIEVIS, el concepto se lleva un paso más lejos en la implementación de una plataforma que permite la compilación sistemática de experiencias STEM, STEAM y xSTEAM (consideradas como "cápsulas STEAM"), la exposición de estas en un museo virtual que además facilita el monitoreo y evaluación de resultados. Adicionalmente, la plataforma permite la organización sistemática y la administración federal (más detalles en [6]).

3.3 Variables específicas de control

Cada experiencia STEAM implica una serie de pasos y artefactos que producen información específica que se refiere a la implementación y su calidad. Para la generación de las métricas de evaluación hay un conjunto de datos que son típicos a cualquier caso, por caso cantidad de asistentes, cantidad de horas de desarrollo, temática, institución que interviene, franja etaria, etc. Paralelamente se identifican otras variables que son propias del tema, lugar y objetivos identificados para la implementación, como el tipo y cantidad de material bibliográfico, existencia y duración de videos, materiales de laboratorio, electrónica, etc. Ambos tipos de variables son determinadas como métricas y se les asocia uno o más indicadores. El proceso de definición de las variables está fuera del alcance de este trabajo, pero basta decir que las generales se hallan implementadas en la plataforma [6], y forman parte de una base de datos que se alimenta de cada nueva "cápsula STEAM" (englobando las xSTEAM), y son usadas para realizar minería de datos.

Las restantes variables deben ser determinadas en cada caso siguiendo el ciclo de vida considerado como parte del protocolo definido, que se presenta con mayor detalle en la sección siguiente.

3.4 Protocolo de adaptación STEAM

El protocolo de desarrollo de experiencias STEAM se encuadra en un modelo de Ciclo de vida para las Cápsulas STEAM y extiende al ciclo de vida del framework desarrollado por LINCIEVIS. La adaptación de los xSTEAM en este contexto da lugar a un protocolo específico. Se entiende por cápsula a un Objeto de Aprendizaje (OA) que conforma la experiencia educativa STEAM; típicamente puede incluir material teórico textual en forma de documentos, videos, imágenes, información de participantes, placas electrónicas, materiales descartables, sonidos, etc.

Dicho ciclo de vida es estable, pero se encuentra en constante revisión y validación para monitorear de esta manera la necesidad eventual de mejora. Para xSTEAM se busca tomar en especial consideración las variables que hacen a los contextos extremos y las restricciones que dificultan el acceso a las experiencias inmersivas STEAM de manera plena. La Figura 1 presenta el modelo de ciclo de vida para las cápsulas STEAM tradicionales [12].

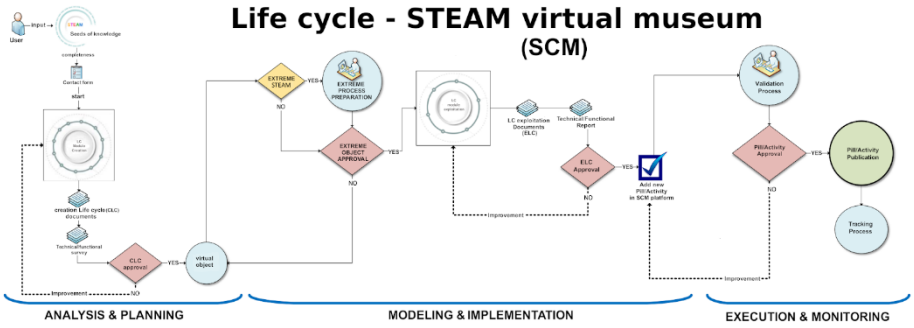


Fig. 1. Ciclo de vida para cápsulas STEAM

Este modelo de ciclo de vida es la esencia de la plataforma Web denominada Museo Virtual STEAM (en adelante MVS). Se trata de una plataforma tecnológica creada para facilitar el acceso a las distintas actividades formativas con lineamiento STEAM. Su objetivo es facilitar el abordaje de actividades colaborativas, mediante la publicación de objetos de aprendizaje denominados "Cápsulas" y la articulación administrativa y de personal a través de la SCA. MVS se constituye como una herramienta tecnológica gratuita de uso general y abierta promovida por la SCA y apoyada por un equipo multidisciplinario de docentes que participan en la construcción del contenido a exponer en cada Cápsula. También se integran colaboradores del área tecnológica para el diseño y construcción del sitio mediante la herramienta WordPress [13].

Finalmente, cabe recordar que el alcance del presente trabajo se encuentra limitado al segmento de los casos comprendidos en contextos educativos restringidos en tecnologías digitales, y la aplicación del análisis conforme el ciclo de vida específicamente diseñado.

4. Caso de uso

Como se ha mencionado, el abordaje sistemático de las cápsulas STEAM derivan en la generación de variables e indicadores. En el caso de xSTEAM los lineamientos son los mismos, salvo que los elementos a ser considerados y monitoreados suelen tener características específicas que no se encuentran en las clásicas experiencias del sector.

4.1 Variables y restricciones

Respecto a las variables xSTEAM son clasificadas, como en los casos estándar del MVS, en globales y específicas. Tal como se mencionó en secciones anteriores, las primeras han sido identificadas y publicadas previamente [5], y las segundas dependen del ciclo de vida específico.

En cuanto a las restricciones, estas se derivan de las condiciones típicas mencionadas anteriormente. La enumeración resulta de una encuesta post-mortem en cada caso y el estudio de los casos involucrados (entre los cuales se halla el caso de uso aquí presentado). Se clasifican aquí en las siguientes categorías:

4.1.A. Brecha docente. En el Docente, los componentes detectados para la brecha se manifiestan por uno o más de los siguientes elementos:

- Dificultad de asumir nuevos roles y/o inseguridades para desarrollar estas prácticas.
- Falta de programas de capacitación previo en este tipo de prácticas mediadas por las tecnologías.
- Confusiones que hay respecto al uso de las TIC (Tecnologías de la Información y Comunicaciones) y al aprendizaje orientado al desarrollo de competencias mediado por tecnologías.

4.1.B. Brecha institucional. Se detectó típicamente una brecha entre la Institución y sus Directivos, esencialmente compuesta por al menos uno de los siguientes:

- Dificultades en adoptar una postura clara respecto a la formación de competencias

STEAM.

- Dificultades en implementar políticas que favorezcan la innovación en las prácticas docentes.
- Dificultad en compatibilizar las prácticas STEAM con las exigencias del sistema educativo formal.

4.1.C. Brecha en los tutores/comunidad educativa. En cuanto a la brecha en tutores/comunidad educativa, se manifiesta como uno o más de los aspectos:

- Diferencias en el grado de involucramiento y/o la dificultad en asumir un nuevo rol que signifique mayor participación en determinadas actividades de aprendizaje orientadas al desarrollo de competencias que realiza el/la estudiante.
- Falta de antecedentes con propias experiencias de enseñanza aprendizaje STEAM.
- Particularidades de las características generacionales.

4.1.D. Brecha de estudiantes. La brecha también se suele manifestar en los estudiantes y se refiere a:

- La forma de comunicarse, particularmente en lo referido al lenguaje (con todo lo que representa en lo semántico) entre estudiantes y docentes.
- Tendencia diversa por temáticas relacionadas particularmente al pensamiento computacional.
- Preferencia por una posición pasiva en el proceso de aprendizaje, con poca/nula

4.1.D. Brecha de contexto. En cuanto a la brecha relacionada con el contexto, se manifiesta como uno o más de los siguientes aspectos:

- Bajo o nulo alfabetismo digital y/o carencias tecnológicas (disponibilidad de dispositivos computacionales, accesibilidad, etc.)
- Grado de compatibilidad con el sistema de valores y expectativas.
- Grado de comprensión del comportamiento generado por la situación, las personas y su cultura.

Tanto en el caso de uso que se presenta como en otras experiencias que se han llevado a la fecha, se han hallado elementos que conforman parte de las restricciones y variables que se han identificado en prácticas xSTEAM, ellas son:

- *nTec, nDisp, nHDisp, nHWeb*: variables ligadas a la determinación del grado de **Acceso a la tecnología**, (cantidad de dispositivos tecnológicos para la experiencia, cantidad de personas por dispositivo, etc.).
- *nPScDoc, nPScDir, nPScCom, nPScEst*: variables del **Contexto sociocultural** implicados en el desarrollo de la actividad desde la perspectiva socioeducativa y cultural, ej. costumbres y usos.
- *nZA, nZD, aLLT, nElect, aCL*: variables del **Contexto educativo,** es decir los factores que intervienen en él y/o los encuentros programados para realizar las actividades de las prácticas STEAM y la perspectiva geográfica de la institución.
- *hPDoc, hPAdm, dPAlu*: variables respecto al **Compromiso de las partes**, se refiere al grado de involucramiento y participación de docentes y estudiantes en las actividades propuestas dentro del proyecto como en la comunidad escolar inserta (es decir, la integración con otras áreas y/o espacios curriculares).
- *nExpAlu, nExpDoc, nExpDir*: medidas de las **Expectativas**, gestionar las expectativas y la vivencia del proceso continúa siendo un factor para tener en

consideración, sobre todo considerando los sesgos o preconceptos sobre la propia temática por desconocimiento y/o por vivencias (experiencias previas).

- *nDExpDoc, nDExpAlu*: valoración de los **Conocimientos previos**, de parte de las partes incluyendo las habilidades y competencias digitales.

Asimismo, algunas reflexiones son de interés al caso xSTEAM en general:

- Es necesario trabajar para compensar las posibles desigualdades materiales, de condiciones de vida, de acceso a las tecnologías y de competencias.
- Es preciso realizar un análisis de riesgos y su respectiva gestión, ante las particularidades de cada caso.
- Es importante analizar la dinámica de las variables conforme a la ejecución del proyecto y, en consecuencia, proponer soluciones y alternativas.
- Sobre el contexto educativo, es importante informar la realidad educativa en pos de un mayor conocimiento del estado de situación actual de los participantes (debilidades y fortalezas).

4.2 Caso de Uso: Trivias Chaco-Jujuy

El caso de uso de esta sección remite a una experiencia xSTEAM desarrollada entre septiembre y noviembre de 2021, con dos escuelas del nivel secundario de las provincias de Chaco (Escuela de Educación Secundaria 129) y Jujuy (Colegio Secundario de Artes N°30 de la localidad de Casira). La organización y gestión para la ejecución de la actividad estuvo a cargo de integrantes del equipo de LINCIEVIS.

Las edades de los estudiantes comprenden desde los 12 hasta los 18 años.

4.2.A. Proyecto xSTEAM: "Escribe tu propia aventura". El proyecto STEAM del caso de uso se denomina "Escribe tu propia aventura". Plantea como desafío armar una historia con recorrido variable (al estilo de "Elige tu propia aventura") utilizando Scratch. Las etapas parten de la construcción de un relato con múltiples alternativas y bifurcaciones, a saber:

- *Paso 1 - Programar una historia secuencial.* Este nivel está dedicado a sentar las bases sobre cómo empezar a programar una historia secuencial en Scratch.
- *Paso 2 - Programando las bifurcaciones*. En este nivel se busca identificar cómo programar caminos alternativos (bifurcaciones) en Scratch.
- *Paso 3 - Actores en Scratch.* En este nivel nos transformamos en "directores" de la historia y buscamos identificar cómo crear personajes para la historia.
- *Paso 4 - Director de escena.* En este nivel se programa el guion de la historia.
- *Paso 5 - Ajustes y ¡Desafío completo!* En este nivel veremos cómo realizar ajustes en nuestra historia y agregar todos los detalles que queramos.

Se busca que los participantes construyan las habilidades fundamentales para programar en Scratch, trabajando en conceptos de representación, abstracción, descomposición y modularización El recorrido debe realizarse a fin de ir acumulando puntos que ayudan al grupo a complementar el puntaje básico obtenido con los pasos básicos; de esta manera compiten por subir en una tabla de posicionamientos respecto a los otros grupos.

Dadas las características de la experiencia xSTEAM, a fin de fortalecer el vínculo cultural entre ambos grupos de estudiantes, se incorporan desafíos que incitan a compartir los usos y las costumbres de cada lugar, a saber:

- *Contacto con la comunidad general. Ir al comercio más cercano y pedir lo más "típico o curioso" del lugar, sacar foto/audio/video, compartirlo describiéndolo.*
- *Contacto con la naturaleza. Sacar foto de un animal del lugar (que no sea mascota), compartirlo describiéndolo, incluyendo sus hábitos y/o características.*
- *Contacto con su hábitat. Identificar una edificación de la ciudad/pueblo/localidad, sacar foto y compartirlo describiéndolo.*

Se realizaron encuentros sincrónicos y asincrónicos, y desafíos con trivias obligatorias. Los encuentros sincrónicos, de 60 a 90 minutos de duración, estuvieron coordinados por tutores de ambas provincias (3 docentes). Se realizaron 10 encuentros semanales, en grupos de trabajo de acuerdo con el ciclo básico/orientado. Los encuentros asincrónicos fueron organizados por los docentes de cada institución, ya que éstos actuaban de tutores. Las actividades obligatorias y los desafíos tenían establecido un tiempo de entrega de 1 a 2 semanas, según el paso al que correspondía.

Finalmente cabe aclarar que inicialmente se propuso realizar las valoraciones por medio de puntuaciones de tipo general y una eventual entrega de premio/reconocimiento a los ganadores.

4.2.B. Resultados de la experiencia. Particularmente en Jujuy, al cuerpo directivo de la institución se sumó el cuerpo docente, conformado por aproximadamente 3 a 4 personas. En Chaco colaboraron el cuerpo directivo y docentes de dos asignaturas.

Los desafíos relacionados con la disponibilidad informática y/o de conectividad a internet, la disponibilidad horaria y el retraso en el desarrollo curricular, hicieron que docentes, directivos y equipo de trabajo xSTEAM idearan un plan fuera de su contexto habitual, el cual entendemos (y de acuerdo con sus propias palabras) fue desarrollado con éxito, habiendo cumplido parte del propósito establecido para la experiencia. Cabe resaltar que tanto estudiantes como docentes lograron apreciar el uso de la tecnología para la educación como una nueva forma de aprender.

Se espera continuar con estas actividades, contemplando eventuales carencias tecnológicas y posiblemente contextos de vida muy diferente a las demás.

5. Conclusiones y Trabajos Futuros

Este trabajo ha introducido la problemática de las actividades STEAM en contextos desfavorables en el ámbito digital y tecnológico. También se ha descrito brevemente el proyecto realizado por LINCIEVIS, que comprende un modelo de ciclo de vida general y una plataforma denominada MVS. Se ha presentado un caso real de condiciones xSTEAM y diferenciado las características e hipótesis que implica este tipo de actividades. También se han presentado las variables derivadas del trabajo y las condiciones determinadas.

A futuro queda la aplicación de las variables y el ciclo de vida extendido para casos xSTEAM, el estudio estadístico de éstas y el seguimiento masivo respecto a STEAM. También resta seguir validando en base a los eventuales hallazgos estadísticos:

- La existencia de una proactividad limitada y/o nula de los docentes respecto a la incorporación de prácticas innovadoras para el desarrollo de competencias STEM.
- La existencia de una brecha en la comunicación entre estudiantes y docentes.
- El grado de receptividad para desarrollar las actividades en modalidad de desafíos, basadas en programación, y según las herramientas que dispongan.
- El nivel de compromiso y entusiasmo de cada participante.

Cabe destacar que la necesidad de un amplio debate, reflexión y sociabilización sobre la innovación en la práctica docente respecto al desarrollo de competencias STEM, las que pueden facilitarse con un adecuado y sistemático control de resultado y calidad. También la evaluación desde la perspectiva de situaciones extremas como las planteadas en este trabajo.

Finalmente queda por seguir validando el modelo de ciclo de vida xSTEAM que se encuentra representado por las variables presentadas aquí, y el protocolo implementado en el ciclo de vida de la plataforma, por lo que es de esperar que siga evolucionando en la medida que se estudien profundamente los modelos derivados por minería y estadística sobre los datos.

Referencias

1. Organización de las Naciones Unidas (ONU) (2022). Objetivos de Desarrollo Sostenible (ODS). www.un.org/sustainabledevelopment, recuperado el 1 de julio de 2023.
2. Ng S. B. (2019). Exploring STEM Competences for the 21st Century. UNESCO int. Bureau for Education. IBE/219/WP/CD/30 REV. pp 53.
3. Quilez Cervero, C. (2020). Revisión teórica e implementación práctica de una secuencia didáctica STEAM basada en diferentes metodologías activas.
4. Moore T. J., y Smith K. A. (2014) Advancing the state of the art of STEM integration. Journal of STEM Education: Innovations and Research, 15(1), 5.
5. López De Luise D., Ruiz Tabarez E. A. (2020). Factores relevantes en la educación STEAM: Desarrollo de Métricas y Modelos Automatizados.
6. López De Luise D. (2020). Museo Virtual STEAM. Tesis presentada para la Especialización en comunicación pública de la Ciencia y tecnología. UBA.
7. Vásquez Giraldo, A. L. (2014). Hacia un perfil docente para el desarrollo del pensamiento Computacional basado en educación STEM para la 108 media técnica en Desarrollo de Software. Universidad EAFIT, Escuela de Ingenierias. Medellín.
8. González H., Kuenzi J. (2012). Science, Technology, Engineering, and Mathematics (STEM) Education: A Primer. Congressional Research Service (CRS) Report for Congress.
9. Ministerio de Ciencia, Tecnología e Innovación (2022). Plan Nacional de Ciencia, Tecnología e Innovación (PNCTI) 2030.
10. Tsupros N., Kohler R., Hallinen J. (2009). STEM education: A project to identify the missing components. Carnegie Mellon, Pennsylvania.
11. Nivela-Cornejo, M. A, Echeverría-Desiderio, S. V., Obter-Agreda, O. E. (2020). Estilos de aprendizajes e inteligencia artificial.
12. Salina C., López De Luise D., Leiva S., Ramirez S., Romero G. R. (2022). A Life-cycle to Improve STEAM Virtual Museum Activities Exposition. 10th International Workshop on Soft Computing Applications 21-23 NOV-2022 Arad, Romania
13. Leiva S. M. (2022). Plataforma Virtual de Educación STEAM. Evaluación Estadística y Minería de Rendimiento. Trabajo Final. Universidad Abierta Interamericana (UAI).

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Aprendizaje y Enseñanza de Programación: El desafío de herramientas de Inteligencia Artificial como ChatGPT

Laura De Giusti[1,3], Gonzalo Lujan Villarreal[3,4],
Eduardo Ibañez[1], Armando De Giusti[1,2]

[1] Instituto de Investigación en Informática LIDI (III-LIDI),
Facultad de Informática, Universidad Nacional de La Plata –
Comisión de Investigaciones Científicas de la Provincia de Buenos Aires
[2] CONICET – Consejo Nacional de Investigaciones Científicas y Técnicas
[3] CICPBA – Comisión de Investigaciones Científicas de la Provincia de Buenos Aires
[4] PREBI – SEDICI – Universidad Nacional de La Plata
{ldgiusti, eibanez, degiusti}@lidi.info.unlp.edu.ar; gonzalo@prebi.unlp.edu.ar

Abstract. En este trabajo se analiza el empleo de herramientas de IA y en particular ChatGPT en la enseñanza y el aprendizaje inicial de programación a nivel universitario. Se discuten cuestiones conceptuales y se sintetiza el análisis a soluciones con ChatGPT de enunciados clásicos que resuelven los alumnos en los cursos de primer año, en un lenguaje imperativo, otro orientado a objetos y un tercero que permite expresar algoritmos concurrentes. El análisis crítico del código generado permite reflexionar sobre la utilidad actual de estas herramientas en el ámbito universitario, las posibilidades que ofrecen y algunos desafíos que plantean, tanto para los alumnos como para los docentes de asignaturas relacionadas con Programación.

Keywords: ChatGPT, Enseñanza inicial, Paradigmas.

1. Introducción

Es interesante iniciar este artículo con la respuesta que da ChatGPT a la pregunta: Podremos reemplazar los programadores utilizando herramientas de Inteligencia Artificial como chatGPT? La respuesta se sintetiza a continuación [1]:

"Es poco probable que las herramientas de Inteligencia Artificial, como ChatGPT, reemplacen por completo a los programadores en un futuro cercano. Aunque los avances en la IA han sido significativos en los últimos años, todavía existen ciertas limitaciones y desafíos que impiden que estas herramientas sustituyan completamente la labor humana en el desarrollo de software.

ChatGPT y otras aplicaciones similares son impresionantes en su capacidad para generar texto coherente y responder preguntas, pero su enfoque actual es más útil para tareas específicas, como proporcionar información, ayudar en la redacción de contenido o incluso asistir en algunas partes del proceso de desarrollo de software.

Sin embargo, hay varias razones por las cuales es poco probable que estas herramientas reemplacen a los programadores:

1. Complejidad de la programación: La programación es una tarea compleja que implica la resolución de problemas, la toma de decisiones y el diseño de algoritmos. Aunque las herramientas de IA pueden ayudar en ciertos aspectos, la

naturaleza creativa y lógica de la programación requiere habilidades humanas.

2. Creatividad e innovación: La programación no se trata solo de seguir instrucciones, sino también de ser creativo e innovador para encontrar soluciones óptimas y eficientes. La creatividad humana es difícil de replicar completamente mediante algoritmos de IA.
3. Interpretación del contexto y la intención: Los programadores no solo deben escribir código, sino también entender el contexto y las intenciones detrás de los requisitos del proyecto. La interpretación y la comunicación efectiva con los clientes o usuarios son tareas en las que la inteligencia humana es esencial.
4. Errores y depuración: La depuración de código y la corrección de errores son aspectos fundamentales del desarrollo de software. La IA podría ayudar en algunas partes, pero identificar y solucionar problemas complejos a menudo requiere habilidades humanas y conocimientos técnicos profundos.
5. Ética y responsabilidad: La programación a menudo involucra tomar decisiones éticas y comprender las implicaciones sociales y legales de las soluciones desarrolladas. Estas cuestiones requieren una comprensión humana más profunda y una consideración consciente.

En lugar de reemplazar a los programadores, las herramientas de IA y la automatización están más orientadas a complementar su trabajo. Pueden ayudar a acelerar ciertas tareas, mejorar la productividad y proporcionar asistencia en el proceso de desarrollo. Sin embargo, la programación seguirá siendo una disciplina en la que la intervención humana y el juicio crítico serán fundamentales durante mucho tiempo."

Esta respuesta es coherente con algunos conceptos que se conocen:

- ChatGPT es una aplicación de chatbot de inteligencia artificial desarrollado en 2022 por OpenAI que se especializa en el diálogo. Se trata de un gran modelo de lenguaje, con más de 150 millones de parámetros y entrenado con un gran volumen de datos (en particular texto) que se ha ajustado con técnicas de aprendizaje tanto supervisadas como de refuerzo. Ha sido pensado inicialmente para realizar tareas relacionadas con el lenguaje, desde la traducción hasta la generación de texto. [2,3,4].
- Desde el punto de vista de la formación de alumnos, en particular en temas de Informática, es importante remarcar la diferencia entre y "formarse para resolver problemas utilizando programación" y "poder generar código en un lenguaje de programación". Naturalmente en las asignaturas iniciales de las carreras de Informática buscamos el primer objetivo [5,6], que consiste en desarrollar aptitudes tales como la modelización de problemas del mundo real, la abstracción en enunciados de los mismos para convertirlos en algoritmos ejecutables en computadora, así como la expresión de soluciones en diferentes lenguajes de programación. Claramente las herramientas de IA como ChatGPT pueden cumplir más adecuadamente con el segundo punto, ya que dado un enunciado especificado correctamente y un lenguaje de programación que esté en su base de conocimiento, es muy posible que genere un código útil. [7,8]
- Las herramientas de IA "tipo ChatGPT" pueden ser útiles para el alumno y el docente en varios aspectos complementarios al aprendizaje y la enseñanza de la programación, tales como sugerencias sobre un código ya desarrollado incluyendo detección de errores o soluciones tentativas a un problema simple que sirvan para

su análisis y perfeccionamiento por el docente y/o el alumno. [9,10]

- Naturalmente en el ámbito profesional (donde el usuario ya está capacitado como programador) y en un lenguaje determinado, estas herramientas pueden acelerar el proceso de desarrollo de software al generar rápidamente código que luego el programador formado puede perfeccionar. [11,12,13]
- Obviamente el alcance y calidad de las soluciones que ofrecen hoy estas herramientas de IA como generadoras de código, está acotado al conjunto de lenguajes que manejen y la variedad de soluciones (y su evolución) que tienen en su base de conocimiento. Como veremos más adelante, si el lenguaje le resulta poco conocido o si la complejidad del problema obliga a combinar recursos de un lenguaje y otro (por ejemplo buscando mejorar el tiempo de respuesta o la eficiencia o el consumo energético) es más difícil "reemplazar un programador" por una herramienta de IA tipo ChatGPT. [14]

2. Análisis de Ejemplos

En la Facultad de Informática de la UNLP las asignaturas Conceptos de Algoritmos Datos y Programas (CADP) y Taller de Programación (TP) son comunes del primer año de los planes de estudio vigentes tanto de las Licenciaturas cortas de Analista.

En la primera de ellas se abordan conceptos iniciales de programación, utilizando programación estructurada, incluyendo temas como el análisis de un problema, la modelización de este, la expresión del algoritmo que lo resuelve, la modularización en el caso de la programación procedural y la definición de las estructuras de datos necesarias para resolver el mismo.

Se discuten en el curso conceptos de importancia en la formación del alumno como la elección correcta de las estructuras de control y de datos, la documentación adecuada del código, su corrección funcional y también su eficiencia relativa a otros modos de implementar la solución.

En esta primera asignatura se utiliza el lenguaje Pascal como una guía para que el alumno pueda implementar los algoritmos y eventualmente probar en una máquina sus soluciones. Si bien Pascal normalmente no es un lenguaje utilizado comercialmente en programación en la actualidad, resulta claro que se lo emplea en el ámbito educativo por ser simple, estructurado, fácil de entender y ejecutable en cualquier modelo de máquina que el alumno disponga (en el aula o en su casa). Por otro lado se busca que el alumno se concentre en la expresión correcta de los algoritmos y no en las complejidades del lenguaje. En el segundo semestre como correlativa de CADP los alumnos cursan Taller de Programación en donde se ven tres paradigmas de programación: imperativo, orientado a objetos y concurrente. En el primer modelo de programación imperativa, se profundizan los conceptos de CADP y se introducen los conceptos de recursividad y estructuras de datos no lineales como árboles; en el segundo modelo, programación orientado a objetos, se presentan los conceptos de clases e instancias, herencia, jerarquía y polimorfismo, y se utiliza el lenguaje de programación Java; finalmente, en el tercer modelo se abordan de manera inicial los conceptos de la programación concurrente, vinculados a la división del problema en procesos que se comunican y sincronizan utilizando memoria compartida o pasaje de

mensajes, y se utiliza el entorno CRME [15] , desarrollado por investigadores del instituto III-LIDI, para implementar los algoritmos en un lenguaje concurrente básico (R-Info). Es importante mostrar en estos ejemplos la sincronización por memoria compartida y por mensajes y su potencial combinación en un problema real.

En este punto la idea de este trabajo ha sido "preguntarle" a ChatGPT cuál es la solución a diferentes problemas planteados y analizar la misma. Este análisis incluye evaluar si existe respuesta, si el código generado es correcto y si tiene en cuenta conceptos de importancia en la formación de los alumnos.

Ejemplo 1: Aquí se abordan temas relacionados con la programación procedural incluyendo análisis del problema, modularización del mismo, utilización de las estructuras de control (while, repeat until, for e if) y las estructuras de datos (registros, arreglos y listas) vistas en el curso.

Usuario>> *La compañía Canonical Llt. desea obtener estadísticas acerca del uso de Ubuntu Linux en La Plata. Para ello, debe realizar un programa que lea y almacene información sobre las computadoras con este sistema operativo (a lo sumo 10.000). De cada computadora se conoce: código de computadora, la versión de Ubuntu que utilizan (18.04, 17.10, 17.04, etc.), la cantidad de paquetes instalados y la cantidad de cuentas de usuario que poseen. La información debe almacenarse ordenada por código de computadora de manera ascendente. La lectura finaliza al ingresar el código de computadora -1, que no debe procesarse. Una vez almacenados todos los datos, se pide:*

- *Informar la cantidad de computadoras que utilizan las versiones 18.04 o 16.04.*
- *Informar el promedio de cuentas de usuario por computadora.*
- *Informar la versión de Ubuntu de la computadora con mayor cantidad de paquetes instalados.*
- *Eliminar la información de las computadoras con código entre 0 y 500.*

Respuesta ChatGPT>> *¡Perfecto! El enunciado del ejercicio es claro. Este programa cumple con los requerimientos del enunciado.*

Primero, se define el tipo de registro `Computadora` para almacenar los datos de cada computadora. Luego, se declara un arreglo `computadoras` para almacenar hasta 10,000 computadoras. El programa utiliza un bucle repetitivo para leer los datos de las computadoras hasta que se ingrese el código -1 o se alcance el límite máximo. Después de la lectura, se procesan los datos para obtener los resultados requeridos.

Finalmente, se muestran los resultados en la salida. Ten en cuenta que este código asume que los datos ingresados son válidos y no realiza una validación exhaustiva. Si deseas agregar verificaciones adicionales, puedes modificar el código según tus necesidades.

Espero que esta solución te sea útil. Si tienes alguna otra pregunta, no dudes en preguntar.

En este momento chatGPT brinda una solución que se puede ver de manera completa en:

https://github.com/gonetil/chatGPT-cadp-taller/blob/main/cadp/solucion1.pas

En dicha solución se puede observar lo siguiente:

- No incluye modularización, concepto fundamental y obligatorio desde el curso inicial de nivelación de la carrera. Modularizar correctamente es una exigencia

para los alumnos en las evaluaciones normales de la asignatura.

- El procesamiento de los datos se realiza mientras se leen los mismos. Si bien esto no es incorrecto desde el punto de la solución, no responde al enunciado que pedía procesar luego de leer los datos.
- La estructura de control utilizada es una estructura postcondicional (repeat until), cuando se esperaba un precondicional (while). Si bien la solución funciona, la elección de la estructura de control más adecuada es uno de los puntos importantes en este tipo de ejercicios para evaluar.
- La eliminación de los datos en el vector se realiza poniendo valores en 0 en las posiciones que se deben eliminar. En este punto es importante aclarar que en la asignatura se trabaja con dimensiones físicas (cantidad máxima de elementos que se pueden almacenar) y lógicas (cantidad real de elementos almacenados) en un vector, realizando corrimientos de los elementos cada vez que uno debe ser eliminado.

Por lo tanto, analizando estos errores desde el punto de vista de los conceptos abordados en la materia se hace un refinamiento sobre el pedido para el mismo ejercicio, pero indicando algunas pautas:

Usuario>> *Ahora quiero ajustar el programa para que utilice Procedures y Functions, y para que primero se cargue la información en un vector y luego se procese dicho vector para resolver los distintos puntos. Para eliminar valores del vector, no se deben poner en cero, sino que se deben realizar los corrimientos necesarios entre posiciones y la correspondiente actualización de la dimensión lógica*

En respuesta ChatGPT brinda una nueva solución que se puede ver de manera completa en:

https://github.com/gonetil/chatGPT-cadp-taller/blob/main/cadp/solucion2.pas

En dicha solución se puede observar lo siguiente:

- La solución no modulariza la lectura de datos. Esto significa que el código para leer los datos está mezclado con el código para generar la estructura de datos. Esto puede hacer que el código sea más difícil de entender, reusar y depurar, y es un concepto que se remarca durante la cursada: la lectura de datos debe ser independiente de la estructura de datos en la cual serán almacenados.
- La solución sigue utilizando una estructura de control postcondicional.
- Se antepone la palabra clave CONST en los parámetros de entrada, indicando así que no van a ser modificados. Si bien existe este concepto en el lenguaje Pascal, es preferible (y así se explica en la asignatura) que los parámetros sean por valor o copia y parámetros por referencia o entrada/salida.
- Al incluir la modularización se recorre la estructura en cada módulo, lo que reduce la eficiencia general de la solución.
- El corrimiento de los valores guardados en el vector no se realiza de manera correcta debido al mal manejo de los índices. Es importante notar que este tipo de errores no es fácil de detectar por alguien que no tiene claros estos conceptos.

Ejemplo 2: este ejemplo aborda los temas relacionados con la programación procedural incluyendo árboles binarios de búsqueda y por ende soluciones recursivas.

Usuario >> *Quiero resolver un ejercicio del módulo Programación Imperativa de la materia Taller de Programación de la Facultad de Informática de la UNLP. El código debe estar en Pascal, y el enunciado dice:*

Escribir un programa que:

a. Implemente un módulo que lea información de ventas de un comercio. De cada venta se lee código de producto, fecha y cantidad de unidades vendidas. La lectura finaliza con el código de producto 0. Un producto puede estar en más de una venta.

Se pide:

i. Generar y retornar un árbol binario de búsqueda de ventas ordenado por código de producto.

ii. Generar y retornar otro árbol binario de búsqueda de productos vendidos ordenado por código de producto. Cada nodo del árbol debe contener el código de producto y la cantidad total vendida. Nota: El módulo debe retornar los dos árboles.

b. Implemente un módulo que reciba el árbol generado en i. y un código de producto y retorne la cantidad total de unidades vendidas de ese producto.

c. Implemente un módulo que reciba el árbol generado en ii. y un código de producto y retorne la cantidad total de unidades vendidas de ese producto.

En este momento chatGPT brinda nueva solución que se puede ver de manera completa en:

https://github.com/gonetil/chatGPT-cadp-taller/blob/main/imperativo

En dicha solución se puede observar lo siguiente:

- La solución no encapsulaba los datos del árbol en un tipo de datos registro dedicado, sino que todos los campos estaban dentro del mismo nodo del árbol. Esto atenta contra la legibilidad del código, y limita su reutilización.
- La solución utiliza una función para generar los nodos del árbol. Si bien es correcto, no resulta claro para el alumno por el modo en que el tema se explica conceptualmente en el curso. VER
- Al igual que las soluciones del ejemplo 1, no se modularizaba la lectura de datos mezclando el código para leer la información con el código que la procesa. Esto puede hacer que el código sea más difícil de entender y depurar, y reduce su reusabilidad.
- Uno de los problemas más importantes es que la solución cometía dos errores graves: por un lado, en uno de los módulos, confundía el tipo de datos del parámetro real con respecto al parámetro formal al invocarlo, lo que implica que el código no funciona correctamente. Por el otro lado, al recorrer el árbol binario, le faltaba evaluar el caso en que encontró el elemento en el árbol (para lo cual debería acumular un valor). Esto significa que el código no sólo no resuelve correctamente lo pedido, sino que este error puede ser particularmente complejo de identificar para un lector no entrenado (como un alumno de los primeros cursos de programación), ya que requiere comprender muy bien cómo

se ejecuta el algoritmo recursivo, en qué punto debería cortarse la recursión y en qué punto debería invocarse recursivamente.

Ejemplo 3: este ejemplo aborda los temas relacionados con la programación orientada a objetos.

Usuario *>> Quiero resolver un ejercicio del módulo Programación Orientada a Objetos de la materia Taller de Programación. El código debe estar en lenguaje Java. El enunciado dice:*

Queremos representar a los empleados de un club: jugadores y entrenadores.

- *Cualquier empleado se caracteriza por su nombre, sueldo básico y antigüedad.*
- *Los jugadores son empleados que se caracterizan por el número de partidos jugados y el número de goles anotados.*
- *Los entrenadores son empleados que se caracterizan por la cantidad de campeonatos ganados.*

A- Implemente la jerarquía de clases declarando atributos, métodos para obtener/modificar su valor y constructores que reciban los datos necesarios.

B- Cualquier empleado debe responder al mensaje calcularEfectividad. La efectividad del entrenador es el promedio de campeonatos ganados por año de antigüedad, mientras que la del jugador es el promedio de goles por partido.

C- Cualquier empleado debe responder al mensaje calcularSueldoACobrar. El sueldo a cobrar es el sueldo básico más un 10% del básico por cada año de antigüedad y además:

- *Para los jugadores: si el promedio de goles por partido es superior a 0,5 se adiciona un plus de otro sueldo básico.*
- *Para los entrenadores: se adiciona un plus por campeonatos ganados (5000$ si ha ganado entre 1 y 4 campeonatos; $30.000 si ha ganado entre 5 y 10 campeonatos; 50.000$ si ha ganado más de 10 campeonatos).*

D- Cualquier empleado debe responder al mensaje toString, que devuelve un String que lo representa, compuesto por nombre, sueldo a cobrar y efectividad.

F- Realizar un programa que instancie un jugador y un entrenador. Informe la representación String de cada uno.

NOTA: para cada método a implementar piense en qué clase/s debe definir el método.

ChatGPT brinda nueva solución que se puede ver de manera completa en: https://github.com/gonetil/chatGPT-cadp-taller/tree/main/objetos.

En dicha solución se puede observar lo siguiente:

- La solución en Java está muy bien. Si bien puede mejorarse, sobre todo por la falta del uso de un método abstracto en la superclase Empleado, la solución es correcta de acuerdo con los objetivos de la materia.
- Se observó que todas las variables de instancia se declaraban de forma protegida (protected), lo que evitaba el uso de getters y setters, pero atentaba contra el encapsulamiento y ocultamiento de información.

Ejemplo 4: este ejemplo aborda los temas relacionados con la programación concurrente.

Usuario >> *Quiero resolver un ejercicio del módulo Programación Concurrente, de la materia Taller de Programación de la Facultad de Informática de la UNLP. El programa debe realizarse en lenguaje R-Info / CMRE. El enunciado dice*

Tres robots deben recorrer el perímetro de su cuadrante, como se indica a continuación: El robot 1 comienza la esquina (2,2) y debe realizar un cuadrante de 6x6 juntando todas las flores que encuentre El robot 2 comienza en la esquina (5,5) y debe realizar un cuadrante de 10x10 juntando todas las flores y los papeles que encuentre El robot 3 comienza en la esquina (9,9) y debe realizar un cuadrante de 7x7 juntando todos los papeles que encuentre Cada robot que finalice su cuadrante deberá avisar al robot fiscalizador. Al recibir el aviso, el robot fiscalizador indicará inmediatamente una calle a la que deberá dirigirse el robot recolector, considerando que el robot que finalizó primero irá a la calle 20, el segundo a la 21 y el tercero a la 22. Cuando los robots recolectores reciben un número de calle, deberán posicionarse en la avenida 1 de dicha calle, y avanzar a lo largo de la calle depositando en cada esquina un papel, una flor o ambos, según lo que cada robot haya juntado. El recorrido finalizará al completar la calle o vaciarse las bolsas.

ChatGPT >> *Lamentablemente, no estoy familiarizado con el lenguaje de programación R-Info mencionado en el enunciado. Como modelo de lenguaje de IA, mi conocimiento se limita a los lenguajes de programación más comunes como Python, Java, C++, entre otros.*

Sin embargo, puedo ofrecerte una solución general utilizando un enfoque pseudocódigo que puedes adaptar al lenguaje de programación R-Info o cualquier otro lenguaje que estés utilizando.

- La solución no está programada en lenguaje R-Info, y por lo tanto no permite evaluar el buen uso de las herramientas del lenguaje. Por ejemplo, qué información se envía al fiscalizador, o si se considera que el envío es asíncrono y la recepción es síncrona. El pseudocódigo propuesto no considera estos elementos de manera explícita.

2.1 Síntesis de los resultados obtenidos de las respuestas generadas por ChatGPT

Analizando cada una de las soluciones a los problemas planteados podemos observar lo siguiente:

Programación Imperativa	Programación Orientada a Objetos	Programación Concurrente
En general la solución funciona, pero no es correcta en conceptos fundamentales para la formación en programación. En CADP dos conceptos importantes son la modularización que involucra el pasaje de parámetros y también la interpretación correcta del enunciado por el alumno. Ambos conceptos no son considerados en soluciones iniciales de chatGPT. En el caso de soluciones recursivas el problema detectado es peor, ya que las soluciones brindadas contenían errores, los cuales en algunos casos son difíciles de detectar por un usuario no entrenado o formado.	En este tipo de problemas las soluciones presentadas por chatGPT son correctas y las que más se aproximan a la manera en que se enseñan los temas en la asignatura.	No existen soluciones en el lenguaje que se utiliza en la materia y las aproximaciones brindadas no pueden evaluarse. Esto refleja una limitación natural de las herramientas de IA utilizadas para generar código: su conocimiento de lenguajes determinados depende de haber sido entrenadas en los mismos.
Estos problemas observados en las soluciones a ejemplos en los distintos paradigmas hacen difícil que un usuario no formado completamente o que se está empezando a formar pueda: ▪ Identificar determinados errores. ▪ Encontrar alternativas mejores (o más eficientes) a la solución presentada. ▪ Asimilar la solución generada por ChatGPT a los conceptos vistos en las asignaturas.		

3. Impacto en la Metodología de Enseñanza y Aprendizaje de Programación

Al momento es muy difícil tener conclusiones sobre el empleo sistemático y planificado de este tipo de herramientas en la Metodología de enseñanza y aprendizaje en las asignaturas iniciales de Programación. De todos modos, hay que considerar que los alumnos emplean (y emplearán cada vez más) estos recursos en la generación de código.

Mencionaremos algunos aspectos en los que se está trabajando para implementarlos en los cursos iniciales de programación en la Facultad de Informática de la UNLP:

A. Mecanismos de interacción con el alumno, utilizando los desafíos del código generado por ChatGPT (o herramientas similares). Entendemos que si el alumno comprende su rol de analizar críticamente la solución que ofrece la herramienta de IA y mejorarla según las pautas que ha aprendido en el curso, se mejora su motivación para el aprendizaje de criterios de optimización de algoritmos. En particular la cátedra trabaja con los alumnos que recursan con una metodología denominada "Autoaprendizaje Guiado" que tiende a un modelo de "Aula Invertida", en la cual este tipo de desafíos encuadra perfectamente.

B. Insistir en aspectos de fondo en la formación en las capacidades propias de un programador presentando cuestiones conceptuales importantes para este nivel de cursos iniciales, como la modularización, la eficiencia comparada de soluciones, el empleo o no de parámetros, la reutilización de patrones de programación, etc. utilizando las soluciones de ChatGPT (o herramientas similares) para destacar estos aspectos, a partir de su análisis crítico y de cómo perfeccionar un código buscando (por ejemplo) mayor legibilidad o eficiencia o facilidad de mantenimiento.

C. Reforzar el tema de interpretación de enunciados y mostrar como enunciados funcionalmente equivalentes pueden interpretarse de modo diferente por una herramienta automática, para lograr que el alumno haga un análisis profundo del problema que se plantea. El tema es crítico en los cursos iniciales de Programación, más allá del empleo o no de herramientas de IA y mostrarle al alumno que NO debe hacer una lectura apresurada de los enunciados y que es necesario que desarrolle capacidad de abstracción sobre el enunciado del problema es de gran importancia. Para esto emplear la comparativa con interpretaciones parciales o incorrectas de enunciados por ChatGPT es útil.

D. Discutir el empleo de estas herramientas en temas especialmente dificultosos para el alumno de los cursos iniciales de programación, tales como Recursividad / Estructuras dinámicas como Listas, Arboles y Grafos y el concepto de eficiencia y calidad en el código que se genera en un lenguaje dado.

E. Analizar los modelos de evaluación práctica que se utilizan en forma sincrónica y asincrónica (en cursos presenciales y a distancia) y considerar el impacto que puede tener el empleo de herramientas "tipo ChatGPT" por los alumnos en las evaluaciones. Aquí hay un trabajo de sensibilización en aspectos éticos para el alumno (que le serán útiles en su vida profesional) y también en el desarrollo de capacidades tecnológicas desde la asignatura para detectar situaciones de "plagio" basadas en el empleo de herramientas como ChatGPT. Al mismo tiempo esto exige que los docentes planteen evaluaciones que permitan valorar los reales aprendizajes del alumno.

F. Insistir en lo conceptual: Que el alumno comprenda que "tener una solución a un problema correctamente especificado" NO es lo mismo que "analizar un problema del mundo real y encontrar un modo de resolverlo en forma eficiente utilizando una computadora". Para esto se trabajan en enunciados incrementales de problemas que conducen al mismo algoritmo y se muestra que no siempre el "código automático" es correcto, porque la fase de análisis del problema es más importante que l generación de código en sí.

4. Reflexiones, conclusiones y líneas de trabajo futuro

En este trabajo se analizaron cuestiones metodológicas para aprovechar constructivamente el empleo de herramientas de IA tipo ChatGPT en el aprendizaje y enseñanza inicial de Programación. Al mismo tiempo se estudiaron casos simples de algoritmos y su expresión en lenguajes utilizados en primer año en cursos de la Facultad de Informática de la UNLP, de modo de ver la potencialidad de ChatGPT para resolverlos correctamente y qué dificultades/errores se encontraron.

Incorporar herramientas de IA en el Aula es un proceso en pleno desarrollo, en todos los niveles educativos y este trabajo nos muestra potencialidades para favorecer el tiempo de aprendizaje del alumno e incrementar la interacción con las tareas que puede realizar durante los trabajos prácticos y fuera del ámbito de las clases.

Al mismo tiempo nos presenta desafíos como los cambios metodológicos, la capacitación de los docentes y la adaptación de los mecanismos de evaluación de modo de asegurar el aprendizaje real del alumno y evitar que el uso de herramientas de IA lo confundan pensando que la "generación de código que resuelve un algoritmo" cubre los aspectos formativos que se buscan en los cursos iniciales de programación.

Quedan abiertas muchas líneas de trabajo, entre las que mencionaremos:

- Ampliación de los casos de estudio, incrementando la complejidad de los enunciados y de los problemas a resolver.
- Considerar aspectos a optimizar en las soluciones (tiempo, eficiencia, consumo).
- Analizar problemas de concurrencia y sus soluciones automáticas y óptimas.
- Ver comparativamente otras herramientas de generación automática de código en los lenguajes utilizados en la formación inicial en Programación.
- Discutir mecanismos de evaluación y autenticación de los trabajos de los alumnos.

Referencias

[1] https://chat.openai.com/ Julio 2023

[2] Augmented intelligence in programming learning: Examining student views on the use of ChatGPT for programming learning R Yilmaz, FGK Yilmaz - Computers in Human Behavior: Artificial Humans, 2023 – Elsevier. journal homepage: www.journals.elsevier.com/computers-in-human-behavior-artificial-humans

[3] ChatGPT for education and research: Opportunities, threats, and strategies. MM Rahman, Y Watanobe - Applied Sciences, 2023 - mdpi.com

[4] Inteligencia artificial al servicio de la educación: Chat GPT para Docentes Guillermo Antón Pardo. Ediciones Rodio- ISBN 9788418794346. Mayo 2023

[5] Programa de Conceptos de Algoritmos, Datos y Programas (2023). Disponible en: https://www.info.unlp.edu.ar/wp-content/uploads/2023/03/CADP-2023.pdf . Último acceso 8 de agosto de 2023.

[6] Programa de Taller de Programación. Disponible en https://www.info.unlp.edu.ar/wp-content/uploads/2023/03/Taller-de-Programacion-Redictado.pdf . Último acceso 8 de agosto de 2023

[7] AI and Machine Learning for Coders: A Programmer's Guide to Artificial Intelligence de Laurence Moroney. Editorial O'Reilly Media ISBN 978-1492078197.

[8] Is ChatGPT the Ultimate Programming Assistant--How far is it?. H Tian, W Lu, TO Li, X Tang, SC Cheung, J Klein, TF Bissyandé. arXiv preprint arXiv:2304.11938, 2023•arxiv.org

[9] GPTutor: a ChatGPT-powered programming tool for code explanation. Eason Chen, Ray Huang, Han-Shin Chen, Yuen-Hsien Tseng, Liang-Yi Li. International

Conference on Artificial Intelligence in Education 2023.

[10] Use chat gpt to solve programming bugs. NMS Surameery, MY Shakor – International Journal of Information technology and Computer Engineering (IJITC) ISSN : 2455-5290. Vol. 3 No. 01 (2023): Dec 2022-Jan 2023

[11] ChatGPT as a Support Tool for Online Behavioral Task Programming Laurent Avila Chauvet, Diana Mejía Cruz & Christian Oswaldo Acosta Quiroz. Sonora Institute of Technology. Available at SSRN …, 2023 - papers.ssrn.com

[12] Pair Programming with ChatGPT: AI-Enhanced Coding for the Modern Developer de Michael D Callaghan.ISBN 979-8394911040

[13] How Coders Can Survive—and Thrive—in a ChatGPT World . 4 Tips for programmers to stay ahead of generative AI Rina Diane Carballar.IEEE Spectrum – Julio 2023.

[14] Augmented intelligence in programming learning: Examining student views on the use of ChatGPT for programming learning. R Yilmaz, FGK Yilmaz – Computers in Human Behavior: Artificial Humans, 2023 – Elsevier. www.journals.elsevier.com/computers-in-human-behavior-artificial-humans

[15] De Giusti, Laura; Leibovich, Fabiana; Chichizola, Franco; Naiouf, Marcelo (2016). Teaching Concurrency and Parallelism Concepts with CMRE. Journal of Computer Science & Technology; vol. 16, no. 2. ISSN 1666-6038.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

VIST: Herramienta para la Visualización del Seguimiento de la Interacción en Trabajo colaborativo en Moodle

Mitchell Vásquez-Bermúdez[1,4] [0000-0001-8157-8549], Cecilia Sanz[2,3] [0000-0002-9471-0008], María Alejandra Zangara[2] [0000-0001-6013-9790], Jorge Hidalgo[1] [0000-0002-9226-4171]

[1] Facultad de Ciencias Agrarias, Universidad Agraria del Ecuador, Avenida 25 de Julio y Pio Jaramillo, Guayaquil, Ecuador

{mvasquez,jhidalgo}@uagraria.edu.ec

[2] Instituto de Investigación en Informática LIDI. Facultad de Informática, Universidad Nacional de La Plata La Plata, Argentina

[3] Investigador Asociado de la Comisión de Investigaciones Científicas de la Prov. de Buenos Aires

{csanz, azangara}@lidi.info.unlp.edu.ar

[4]Facultad de Ciencias Matemáticas y Físicas, Universidad de Guayaquil, Cdla. Salvador Allende, Guayaquil, Ecuador

mitchell.vasquezb@ug.edu.ec

Resumen. En la actualidad se observan esfuerzos por enriquecer los entornos virtuales de enseñanza y aprendizaje con herramientas para el seguimiento del trabajo colaborativo. En particular, estas herramientas también pueden aportar a identificar la presencia cognitiva y social de los estudiantes. Este trabajo presenta la herramienta VIST que integra meta-anotaciones basadas en las presencias del Modelo CoI y el monitoreo, a través de gráficos en los foros de un entorno de enseñanza y aprendizaje particular, Moodle. Se ha evaluado la usabilidad de VIST en un caso de estudio con estudiantes universitarios. Entre los principales resultados obtenidos con la herramienta VIST, se demostró que la incorporación de meta-anotaciones permite a los docentes y estudiantes monitorear las interacciones de colaboración, y disponer de indicadores relacionados con las presencias del modelo CoI para la toma de decisiones. También se describen algunos aspectos que requieren mejora, que serán el punto de partida para futuros trabajos.

Keywords: Meta-anotaciones, Modelo de Comunidad de Indagación (CoI), Indicadores, Visualización del Seguimiento de trabajo colaborativo.

1. Introducción

El aprendizaje, según el modelo de la Comunidad de Indagación (CoI), se produce a través de la interacción de tres elementos que son: la presencia cognitiva, la presencia social y la presencia docente [1]. A pesar de la amplia aplicación del modelo CoI, su evaluación sigue siendo un reto. La manera de evaluar CoI consiste en el análisis de contenido cualitativo que se realiza manualmente, así como el análisis estadístico de los cuestionarios, que consumen mucho tiempo y, sobre todo, tienen lugar después de la finalización del curso [2]. En el proceso de aprendizaje colaborativo están inmersos varios elementos que son esenciales para su realización y para obtener mejores resultados de colaboración y aprendizaje, es deseable contar con una herramienta tecnológica que permita centralizar las necesidades del docente y del estudiante,

además como resultado lograr un mejor desempeño [3]. Una de las principales aplicaciones utilizadas en cursos impartidos en línea en la educación superior son los Sistemas de Gestión del Aprendizaje (LMS) tales como Moodle y Blackboard, donde los debates asíncronos en línea desempeñan un papel importante.

Las actividades de debate asíncrono tienen como objetivo que los estudiantes compartan información y puntos de vista interactuando con sus compañeros. En los sistemas tipo LMS se encuentran comúnmente, una serie de herramientas basadas en la web para apoyar actividades y procedimientos de gestión de cursos. Sin embargo, aún no se observan herramientas de seguimiento del trabajo colaborativo en los foros [4] basados en el Modelo CoI, que faciliten la visualización en forma gráfica de los datos de interacción en una actividad colaborativa, y que aporten a incrementar la colaboración entre los participantes en el proceso de aprendizaje colaborativo.

En este trabajo se presenta la herramienta VIST propuesta por los autores, que propone la visualización mencionada y que se integra a Moodle. Además, se presenta la evaluación de VIST mediante pruebas de usabilidad en un caso de estudio con la participación de estudiantes en un trabajo colaborativo. Para ello se utilizó el método de prueba de usabilidad conocido como la Escala de Usabilidad del Sistema (SUS)[5].

Este artículo está estructurado de la siguiente manera: la sección 2 presenta trabajos relacionados, en los que se muestran algunas herramientas vinculadas al seguimiento del aprendizaje colaborativo; en la sección 3, se describe la herramienta de visualización propuesta integrada en Moodle y sus características; y en la sección 4, se presenta la utilización en un caso de estudio. Finalmente, se detallan los resultados y las conclusiones.

2. Trabajos Relacionados

Papanikolaou et al. [6] presentan Inspire, una herramienta para visualizar indicadores específicos sobre el comportamiento de las interacciones de los estudiantes, donde los usuarios conforman una comunidad online teniendo uno o más roles: estudiantes, tutores, autores, revisores. Coelho y Braga [7] presentan la herramienta ColMiner, que proporciona la posibilidad de calcular la relevancia de los comentarios publicados por los participantes, con el fin de apoyar la gestión de las comunicaciones. Zhu et al. [8] presentan un estudio que combina el aprendizaje por indagación y aprendizaje colaborativo asistido por computadora en un curso para resolución de problemas y pensamiento crítico, además realizaron un análisis de entropías y minería de patrones secuenciales del trabajo de los participantes en una actividad relacionada con la Ley de Ohm en un entorno de aprendizaje colaborativo basado en la simulación. Martínez et al. [9] definen una evaluación de las interacciones de los participantes mediante la captura de los eventos y su procesamiento, para modelar el estado de la interacción, a partir de un programa que delinea el contenido de las interacciones para almacenarlas y evaluarlas en términos computacionales de forma genérica. Hussain et al. [10] proponen un método para el seguimiento de las actividades de los estudiantes en Moodle con el objetivo de predecir su rendimiento académico, utilizaron varias técnicas de clasificación y agrupación de aprendizaje automático para extraer el patrón de los datos de los estudiantes durante la realización de un curso de Moodle. Kuosa et

al.[11] presentan TUT LA Tool, una herramienta como complemento para el LMS Moodle para la visualización interactiva de la actividad de los estudiantes a partir de datos de usuarios registrados y proporciona información valiosa sobre el proceso de aprendizaje y la participación de los estudiantes en un curso online.

Este trabajo avanza sobre la propuesta de una serie de indicadores vinculados a los conceptos de Comunidad de Investigación (CoI) para evaluar la participación y presencia de los estudiantes en procesos colaborativos en línea, tomando en cuenta las tres presencias: presencia social, presencia docente y presencia cognitiva. Además, se propone el desarrollo de una herramienta tecnológica que posibilite la visualización de los indicadores planteados y permita el seguimiento de las actividades colaborativas de los estudiantes. Una contribución importante en este sentido es la integración de esta herramienta en el LMS Moodle, lo que permitirá a los docentes evaluar con mayor precisión el nivel de interacción y construcción del conocimiento de los estudiantes en procesos colaborativos en línea, analizando los tipos de presencia. Además, facilitará a estudiantes y docentes la toma de decisiones para mejorar estos procesos.

3. Propuesta de una herramienta VIST en Moodle

En esta sección se presentan los principales componentes de la herramienta VIST, que se basa en los indicadores que se presentan en la siguiente sección. Como se muestra en la Fig. 1, la arquitectura de la herramienta integrada en Moodle consta de dos módulos principales: (1) el módulo de integrado en herramientas colaborativas como el foro y (2) el módulo integrado para la visualización de datos.

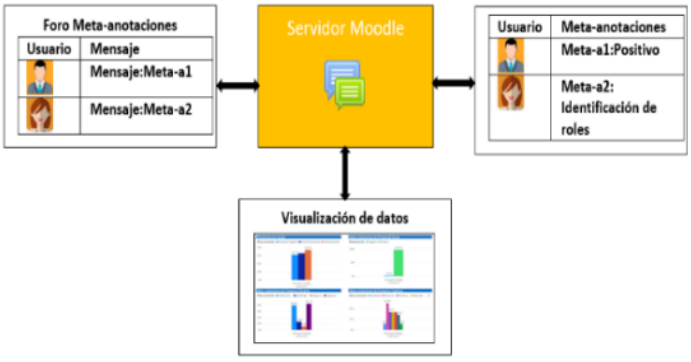


Fig. 2. Arquitectura de VIST en *Moodle*.

3.1 Módulo integrado en la herramienta colaborativa foro

La herramienta VIST se fundamenta en el modelo de CoI y utiliza los indicadores de interacción colaborativa presentados por Zangara y Sanz [12]. Estos indicadores están basados en los tipos de mensajes que intercambian los participantes de un grupo de trabajo colaborativo y se han utilizado como referencia para el diseño de nuevos indicadores relacionados con las tres presencias: la presencia social, la presencia

docente y la presencia cognitiva. En la tabla 1 se muestra los indicadores que están vinculados con cada tipo de presencia.

Tabla 10. Indicadores de los tipos de Presencia

Indicadores propuestos para el seguimiento de un proceso colaborativo		
Presencias	**Nivel 1**	**Nivel 2 (Ejemplos de Indicadores)**
Presencia Social	Afectivo	• Positivo • Negativo
Presencia Docente (según provenga el aporte del docente) o Metacognitiva (si el aporte lo realiza el estudiante)	Organización	• Identificación de roles • Distribución de tareas por roles • Manejo del tiempo • Organiza tareas
Presencia cognitiva	Contenido	Etiquetado por temas abordados Estas etiquetas pueden ser definidas ad-hoc por los docentes antes de iniciar con la consigna de trabajo. Así los participantes pueden meta-anotar sus aportes describiendo el contenido. Ejemplo para un caso de diseño de redes: • Hardware de red • Topología de red • Distribución de equipos • Costos de red • Enlace de red • Ventaja de red

La herramienta VIST utilizada en Moodle para el seguimiento de trabajo colaborativo se implementó con la integración de indicadores propuestos basados en "meta-anotaciones", para que el participante realice su contribución, con esta representación, el estudiante puede elegir, desde una lista de metadatos (indicadores de Nivel 2) aquella que mejor represente la intención de su colaboración, para luego completar su contribución introduciendo texto libre[13]. Los metadatos seleccionados por los participantes durante sus interacciones son almacenados en la base de datos. Inicialmente, el administrador de Moodle ingresa y configura de forma manual los metadatos de Presencia Social y Presencia Docente. Posteriormente, el docente agrega las etiquetas de Presencia Cognitiva según el tema abordado. Además, en Moodle es posible agregar, editar y modificar los metadatos, esto se puede configurar según el contexto de trabajo. Para su implementación en las interfaces del foro se incorporaron los indicadores en forma de etiquetas específicas para cada Presencia CoI y con la opción de buscar la meta-anotación para completar la participación en un área de texto del mensaje. Para proporcionar a los estudiantes la visualización de los distintos

indicadores, se decidió agruparlas por el Nivel 1 en listas desplegables por indicador. En la Fig. 2 se muestra un ejemplo de la interfaz de un foro.



Fig. 2. Foro con las meta-anotaciones

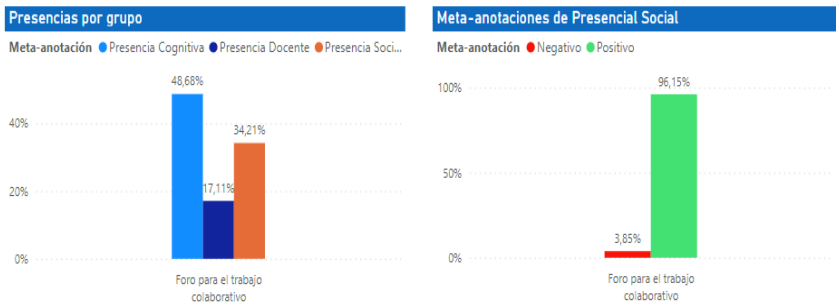
Módulo integrado para la visualización de datos

La herramienta VIST integra a Moodle una visualización de los datos [14] relacionados a los indicadores propuestos para el seguimiento en la herramienta colaborativa de foro. Esta integración muestra, de forma gráfica, la interacción de los participantes en el foro. Como se mencionó anteriormente, la visualización de datos se basa en un conjunto de metadatos que fueron definidos a partir de los indicadores de seguimiento propuestos. Esta herramienta permite mostrar utilizando una representación gráfica, los indicadores que intervienen en el seguimiento de un trabajo colaborativo.

La visualización de datos permite al docente realizar el seguimiento del trabajo colaborativo, así como la capacidad de seleccionar y supervisar a un participante en particular. La visualización de datos también posibilita a los estudiantes supervisar su participación en la actividad colaborativa y, por ende, fomentará la reflexión y autoevaluación del aprendizaje.

Se ofrece una interfaz web que muestra las interacciones de forma gráfica en la herramienta foro. El seguimiento de las actividades se muestra por los grupos de estudiantes y los indicadores de la interacción realizada en el trabajo colaborativo.

En la Fig. 3, se presenta un gráfico dinámico de la contribución de los estudiantes en la actividad colaborativa de foro; donde se aprecia la densidad de interacción por tipo de meta-anotaciones.



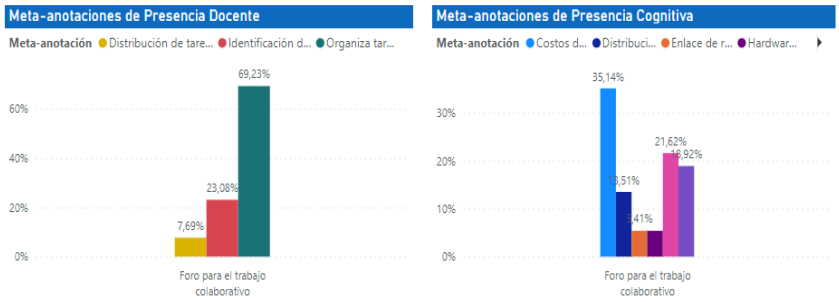


Fig. 3. Densidad de interacción por tipo de meta-anotación en foro

4. Evaluación y resultados

4.1 Estudio de caso

Durante el segundo semestre del 2022, en el curso de " Redes de Computadoras I " en la Universidad Agraria del Ecuador (UAE), se llevó a cabo un estudio de caso para evaluar la efectividad de la herramienta VIST para el seguimiento del trabajo colaborativo en línea. El objetivo principal de esta asignatura es enseñar los fundamentos de las diferentes tecnologías existentes en el campo de redes de computadoras y cómo seleccionarlas de acuerdo con criterios técnicos para proporcionar soluciones eficientes y prácticas a las necesidades de comunicación actuales de la sociedad.

El curso consta de seis horas semanales con tres horas teóricas y tres horas prácticas, se imparte de forma presencial, con una matrícula de 10 estudiantes de educación superior, correspondientes al 60% de género masculino (6), mientras que el 40% de género femenino (4). En la parte práctica del curso, los estudiantes participaron en una actividad colaborativa en línea en el LMS *Moodle*, en la que se les pidió que discutieran y analizaran un caso de estudio de diseño de redes LAN.

Durante las dos semanas que duró la actividad colaborativa en el foro, los estudiantes utilizaron la herramienta VIST para realizar meta-anotaciones en las discusiones y comentarios de sus compañeros. Además, visualizaron los gráficos estadísticos basados en los indicadores propuestos del modelo CoI con el fin de analizar su participación y la de sus compañeros en el proceso.

Se evaluó la usabilidad de la herramienta VIST en el desarrollo de la interacción colaborativa de los estudiantes a través de un cuestionario SUS, que se aplicó al final de la actividad. Los resultados del estudio de caso se utilizarán para determinar la usabilidad y el valor potencial de la herramienta VIST en futuros cursos y actividades colaborativas en línea en la UAE.

4.2 Evaluación de usabilidad

Con el fin de evaluar la usabilidad de la herramienta VIST en *Moodle*, es necesario evaluar las opiniones de los usuarios para medir el nivel de usabilidad. Para llevar a

cabo evaluaciones basadas en las opiniones de los usuarios, se pueden utilizar pruebas de usabilidad. Estas pruebas se enfocan en la experiencia del usuario al utilizar un sistema, por lo que resultan muy útiles. En esta investigación, el método de prueba de usabilidad utilizado es el cuestionario SUS, que fue adaptado del original, como se muestra en la tabla 2. Las respuestas están en escala de Likert entre 1 y 5, siendo 1 total desacuerdo y 5 totalmente de acuerdo.

Tabla 2. Cuestionario SUS adaptado

No.	Cuestionario SUS	Puntuación
1	Me gustaría volver a usar las herramientas colaborativas que incorporan las meta-anotaciones y visualizaciones	1-5
2	Me pareció que las meta-anotaciones y visualizaciones son muy complejas	1-5
3	Me pareció fácil de usar las meta-anotaciones y visualizaciones en las herramientas colaborativas	1-5
4	Necesité ayuda de alguien que sabía usar las meta-anotaciones y visualizaciones en las herramientas colaborativas	1-5
5	Me parece que las diferentes funciones incorporadas a las herramientas colaborativas están bien integradas	1-5
6	Pensaba que las meta-anotaciones y visualizaciones tenían muchas inconsistencias	1-5
7	Me imagino que la mayoría de la gente aprendería a usar las meta-anotaciones y visualizaciones muy rápido	1-5
8	Me parece que las meta-anotaciones y visualizaciones son muy complicadas de usar	1-5
9	Me sentí cómodo usando las herramientas colaborativas que incorporan las meta-anotaciones y visualizaciones	1-5
10	Tuve que aprender muchas cosas para poder usar las meta-anotaciones y visualizaciones	1-5

Para el cálculo del resultado de la evaluación se realizaron los siguientes pasos con las respuestas de los encuestados:

a) En las preguntas impares, es decir: 1, 3, 5, 7 y 9, la puntuación dada por el encuestado se resta en 1.

$$SUS = \Sigma Px - 1 \quad \textbf{(1)}$$

Donde Px es el número de preguntas impares

b) En las preguntas 2, 4, 6, 8 y 10, las puntuaciones dadas por los encuestados se restan en 5.

$$SUS = \Sigma 5 - Pn \quad \textbf{(2)}$$

Donde Pn es el número de preguntas pares.

c) A continuación, se suman los resultados de la conversión de cada encuestado y se multiplican por 2,5 para obtener una puntuación entre 0 y 100.

$$(\Sigma \text{ impares} - \Sigma \text{ pares}) \times 2{,}5. \quad \textbf{(3)}$$

Una vez conocida la puntuación de cada encuestado, el siguiente paso consiste en hallar la puntuación media sumando todos los resultados de puntuación y dividiéndolos por el número de encuestados. Este cálculo puede verse en la siguiente fórmula:

$$X = \frac{\Sigma x}{N}. \quad (4)$$

Descripción:
X = Puntuación media
Σx =Número de puntuaciones de la escala de usabilidad del sistema
N = Número de encuestados

A partir de estos resultados, se obtiene un valor medio de la evaluación de todas las puntuaciones de los encuestados. Para determinar los resultados de la calificación se observa a partir del rango percentil (puntuación SUS), que tiene un grado de evaluación compuesto por A, B, C, D y E. En general, la determinación de los resultados de la evaluación a partir del rango percentil de la puntuación SUS se basa en el cálculo de las evaluaciones realizadas por los usuarios. En la tabla 3 se muestra el rango percentil.

Tabla 3. Rango percentil

Grado	Descripción
A	Puntuación >= 80,3
B	Puntuación >= 74 y <80,3
C	Puntuación >=68 y 74
D	Puntuación >= 51 y <68
E	Puntuación <51

La tabla 4 muestra los resultados obtenidos del cálculo de la puntuación de la escala de usabilidad del sistema (SUS) para cada encuestado.

Tabla 4. Resultados del cuestionario SUS

Participantes	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Valor SUS
P1	4	2	3	1	4	2	5	3	5	2	**77,5**
P2	4	2	5	3	5	2	5	1	4	2	**82,5**
P3	4	2	5	2	4	2	5	2	4	3	**77,5**
P4	4	2	5	2	4	2	5	1	4	2	**82,5**
P5	5	2	5	3	5	2	5	3	4	2	**80,0**
P6	5	3	5	3	5	2	4	2	3	3	**72,5**
P7	5	3	5	3	5	1	4	3	5	2	**80,0**
P8	4	3	5	2	5	2	5	2	3	2	**77,5**
P9	5	1	5	2	5	2	5	1	3	1	**90,0**
P10	4	2	3	1	5	2	4	1	4	2	**80,0**
Promedio	**4,4**	**2,2**	**4,6**	**2,2**	**4,7**	**1,9**	**4,7**	**1,9**	**3,9**	**2,1**	**80,0**

A partir de la información proporcionada en la Tabla 4, se puede inferir que la propuesta realizada que utiliza las herramientas con la incorporación de meta-anotaciones fue considerada con buena usabilidad por parte de los estudiantes. Esto se

basa en los resultados de los análisis obtenidos por medio de la escala de SUS, tras la recopilación de datos a través de un cuestionario. Los resultados indican un promedio de 80 en la escala de usabilidad, lo que corresponde a un grado B y se considera un rango aceptable. En base a estos resultados, se puede afirmar que la herramienta VIST tiene buena usabilidad para su uso por los estudiantes.

5. Conclusiones y trabajo futuro

Este artículo presenta la herramienta VIST que incorpora nuevas funcionalidades a la herramienta foro del Moodle, así como la visualización grafica del seguimiento de actividades colaborativas, tomando en cuenta las meta-anotaciones de los participantes. Los resultados del estudio de caso sugieren que la herramienta VIST es fácil de usar. Esto podría facilitar su integración en el desarrollo de actividades colaborativas, brindando estrategias para mejorar la comunicación y el trabajo colaborativo entre los estudiantes.

Aunque la herramienta VIST cuenta con varias funcionalidades útiles para el seguimiento y visualización del trabajo colaborativo en Moodle, se han identificado algunas limitaciones que deben ser abordadas en futuros trabajos. Los participantes han manifestado la necesidad de la importancia de la integración con otras herramientas de comunicación y colaboración, como chat o videoconferencia, para enriquecer las interacciones y facilitar la comunicación en tiempo real. Además, se debe tener en cuenta que la herramienta VIST está diseñada para el contexto académico de la educación superior y podría no ser aplicable a otros contextos. Este aspecto podría ser analizado con otros casos de estudio.

También, sería interesante explorar posibles integraciones de la herramienta VIST con otras herramientas de Moodle o de terceros para ampliar aún más sus capacidades. Aún resta evaluar el impacto de la herramienta propuesta en los procesos colaborativos, de qué manera aporta de acuerdo con la percepción de estudiantes y docentes. Finalmente, sería útil investigar cómo la herramienta VIST podría integrarse con la inteligencia artificial para proporcionar una gestión más automatizada de las etiquetas y un análisis más sofisticado de los datos.

Referencias

1. Rourke, L., Anderson, T., Garrison, D.R., Archer, W.: Assessing Social Presence in Asynchronous Text-based Computer Conferencing. J. Distance Educ. / Rev. l'ducation Distance. 14, 50–71 (1999).
2. Kovanovíc, V., Joksimovíc, S., Waters, Z., Gašević, D., Kitto, K., Hatala, M., Siemens, G.: Towards automated content analysis of discussion transcripts: A cognitive presence case. ACM Int. Conf. Proceeding Ser. 25-29-April-2016, 15–24 (2016). https://doi.org/10.1145/2883851.2883950.
3. Agredo Delgado, V., Ruiz, P.H., Collazos, C.A., Fardoun, H.M., Noaman, A.Y.: Software tool to support the improvement of the collaborative learning process. In: Communications in Computer and Information Science. pp. 442–454. Springer Verlag (2017). https://doi.org/10.1007/978-3-319-66562-7_32.

4. Delgado Rodríguez, R., John Vásquez-Bermúdez, M., Washington, J., Larrea, H., Rolando López López, R., Rodríguez, D., Vásquez-Bermúdez, R., López, &: Seguimiento de trabajos colaborativos en plataformas de aprendizaje estudiantil mediante herramientas de gestión informática. Rev. Conrado. 18, 77–87 (2022).
5. Usability: System Usability Scale (SUS) | Usability.gov, https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html, last accessed 2023/03/29.
6. Papanikolaou, K.A.: Constructing interpretative views of learners' interaction behavior in an open learner model. IEEE Trans. Learn. Technol. 8, 201–214 (2015). https://doi.org/10.1109/TLT.2014.2363663.
7. Neto, L.E.C., Braga E Silva, G.: ColMiner: A tool to support communications management in an issue tracking environment [ColMiner: Uma ferramenta de apoio ao gerenciamento das comunicações em um ambiente de issue tracking]. ACM Int. Conf. Proceeding Ser. 392–399 (2018).
8. Zhu, G., Xing, W., Popov, V.: Uncovering the sequential patterns in transformative and non-transformative discourse during collaborative inquiry learning. Internet High. Educ. 41, 51–61 (2019). https://doi.org/10.1016/J.IHEDUC.2019.02.001.
9. Martínez, A., Dimitriadis, Y., Rubia, B., Gómez, E., Garrachón, I., Marcos, J.A.: Studying Social Aspects of Computer-Supported Collaboration with a Mixed Evaluation Approach.
10. Hussain, M., Hussain, S., Zhang, W., Zhu, W., Theodorou, P., Abidi, S.M.R.: Mining moodle data to detect the inactive and low-performance students during the moodle course. ACM Int. Conf. Proceeding Ser. 133–140 (2018). https://doi.org/10.1145/3291801.3291828.
11. Kuosa, K., Distante, D., Tervakari, A., Cerulo, L., Fernández, A., Koro, J., Kailanto, M.: Interactive Visualization Tools to Improve Learning and Teaching in Online Learning Environments. Int. J. Distance Educ. Technol. 14, 1–21 (2016). https://doi.org/10.4018/IJDET.2016010101.
12. Zangara, M.A., Sanz, C.: Trabajo colaborativo mediado por tecnología informática en espacios educativos. Metodología de seguimiento y su validación Collaborative work mediated by computer technology in educational spaces. Monitoring methodology and its validation Esta obra se distribuye bajo Licencia Creative Commons CC-BY-NC 4.0 Resumen. Rev. Iberoam. Tecnol. en Educ. y Educ. en Tecnol. 8–20 (2020). https://doi.org/10.24215/18509959.25.e01.
13. Yanacón-Atía, D., Costaguta, R., De Los, M., Menini, A.: Indicadores colaborativos individuales y grupales para Moodle Individual and group collaborative indicators for Moodle. (2018).
14. 14. Vásquez-Bermúdez, M., Sanz, C., Zangara, M.A., Hidalgo, J.: Visualization Tools for Collaborative Systems: A Systematic Review. 107–122 (2021). https://doi.org/10.1007/978-3-030-88262-4_8.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Aplicaciones educativas y realidad aumentada: Un mapeo sistemático de literatura

Paulina Giménez-Suárez[1], Rosanna Costaguta[1] y María de los Ángeles Menini[1]

[1] Universidad Nacional de Santiago del Estero, Instituto de Investigación en Informática y Sistemas de Información, Avda. Belgrano (S) 1917, 4200 Santiago del Estero, Argentina
paulitatipgs@gmail.com, {rosanna, marameni}@unse.edu.ar

Resumen. Este artículo presenta un mapeo sistemático de literatura vinculado con la creación y uso de aplicaciones que disponen de recursos aumentados en contextos educativos. Las fuentes de exploración consideradas fueron: Researchgate, Google Scholar, Sedici, SpringerLink, Dialnet, ScienceDirect, Mendeley, ACM Digital Library y ACS Publications. La búsqueda se realizó teniendo en cuenta las producciones publicadas en español o en inglés, entre los años 2015 y 2023. Inicialmente se obtuvieron 2500 artículos que, luego de concretar las acciones propias de la metodología elegida, se redujeron a 23. Esto permitió identificar y analizar 26 aplicaciones. Las principales características reconocidas fueron: mayoritariamente se basan en el uso de dispositivos móviles y se crearon usando Unity o Vuforia, se abocan a diversas áreas de conocimiento, y se aplican casi por igual a los niveles primario, secundario y universitario. Además, la experimentación con estudiantes reales y el análisis de los resultados obtenidos no se incluye en la mayoría de los artículos, y es casi nula la consideración de aspectos de usabilidad. Finalmente, cabe resaltar la necesidad de desarrollar aplicaciones educativas que dispongan de recursos aumentados y estén especialmente destinadas a estudiantes en contextos colaborativos.

Palabras clave: realidad aumentada, aprendizaje colaborativo soportado por computadora, aplicaciones educativas, usabilidad

1. Introducción

En los últimos tiempos, los avances tecnológicos y de comunicación han permitido que las técnicas y herramientas de educación evolucionen mejorando la calidad de enseñanza y de aprendizaje. Una muestra de tal evolución es la incorporación de herramientas de realidad aumentada (RA) en los procesos de enseñanza y de aprendizaje, que han permitido al estudiante mejorar su participación y motivación, y también facilitado su comprensión de conceptos abstractos, aumentando su imaginación y curiosidad [1].

Azuma [3] define la RA como una variación de los entornos virtuales, donde los objetos reales se combinan con otros virtuales tridimensionales que contienen información, y donde el usuario interactúa en tiempo real. La RA logra integrar al usuario en dos mundos, el real y el computacional [2].

Rigueros [4] afirma que la implementación, el uso y el aprovechamiento de la RA actualmente es notoria, y que esto quizás se deba a que en cualquier parte del mundo es posible acceder a una conexión de internet, y a que los dispositivos inteligentes tales como celulares, tabletas, cámaras, gafas, entre otros, cuentan con el software y hardware necesario. Sin lugar a duda, la RA fue adoptada con éxito por diferentes áreas,

como son: economía, medicina, cultura, etc., y en los últimos años, también por el sector de la educación [5].

Según Yuen *et al.* [6] hay cinco tipos de aplicaciones educativas de RA: aprendizaje basado en el descubrimiento (ABD), modelado de objetos (MO), libros (LI), aprendizaje basado en juegos (ABJ) y entrenamiento de habilidades (EH).En el caso de ABD las aplicaciones brindan a los estudiantes información adicional sobre un lugar del mundo real u objeto de interés, con MO utilizan objetos modelados en 3D para permitir a los estudiantes visualicen cómo se vería un objeto dado desde diferentes puntos de vista, y también les permiten explorar las propiedades físicas y ejecutar interacciones entre objetos, con LI pueden ver personajes y objetos 3D con el uso de marcadores en un libro real, con ABJ pueden usar juegos que logren simplificar o reforzar conceptos complejos, y finalmente, con EH es posible capacitar a las personas en tareas específicas (lo cual es útil para efectuar entrenamientos).

Por otro lado, los docentes también han evolucionado en sus prácticas a fin de mejorar los procesos de enseñanza y de aprendizaje. Una muestra de tal evolución es la adopción del Aprendizaje Colaborativo Soportado por Computadora (ACSC). Según Costaguta *et al.* [7] el ACSC se usa para designar un contexto en el que las experiencias de aprendizaje colaborativo describen situaciones que ocurren en un grupo, esperando ciertas formas de interacción entre sus integrantes, promoviendo mecanismos de aprendizaje e incluyendo, necesariamente, el uso de medios computacionales. Su principal característica es que incorpora los beneficios asociados con un soporte tecnológico, particularmente útil para este tipo de aprendizaje, gracias al cual los participantes pueden contribuir desde cualquier lugar y en cualquier momento.

Actualmente, existen aplicaciones colaborativas que utilizan RA y que acarrean beneficios como: fomentar el trabajo en equipo, propiciar el uso compartido de información, dudas y conocimiento, favorecer el intercambio de ideas y opiniones, todo lo que lleva a un mejor nivel cognitivo [1].

Mediante un mapeo sistemático de literatura, este trabajo busca responder determinadas preguntas de investigación referidas al uso de la RA en aplicaciones educativas. Cabe recordar que un mapeo sistemático de literatura es una revisión rigurosa, ordenada y auditable de publicaciones científicas existentes a fin de dar respuesta a las preguntas de investigación que se hayan formulado. Este documento se estructura en cuatro secciones. En la sección 2 se describen los pasos metodológicos ejecutados para concretar este mapeo sistemático. En la sección 3 se da respuesta a las preguntas de investigación especificadas en este trabajo. Finalmente, en la sección 4, se enuncian algunas conclusiones y líneas de trabajo futuro.

2. Metodología

El mapeo sistemático de literatura se concretó mediante la ejecución de los cinco pasos propuestos por Petersen *et al.* [8]: (a) definición de las preguntas de investigación, (b) búsqueda de estudios primarios, (c) selección de artículos para su inclusión y exclusión, (d) codificación de resúmenes y (e) extracción de datos y mapeo de estudios. Estos pasos se documentan a continuación.

a) Definición de las preguntas de investigación. Las preguntas se definieron a fin de recolectar información acerca del desarrollo y uso de aplicaciones que incluyan RA en los procesos de enseñanza y de aprendizaje. Las preguntas formuladas fueron:

P1: ¿Qué aplicaciones educativas con RA existen?

P2: ¿En qué consisten tales aplicaciones?

P3: ¿Cuáles son sus características principales?

P4: ¿Se aplicaron en contextos de ACSC?

b) Búsqueda de estudios primarios. La búsqueda de antecedentes se realizó de manera automática mediante los motores de búsqueda de las fuentes consultadas: Researchgate, Google Scholar, Sedici, SpringerLink, Dialnet, ScienceDirect, Mendeley, ACM Digital Library y ACS Publications. Se tomaron como válidos artículos abocados al desarrollo y/o uso de una aplicación que incluyera recursos aumentados en el campo de la educación, en idioma inglés o español. Las palabras clave de búsqueda fueron obtenidas a partir de la identificación de conceptos relevantes para la creación y uso de aplicaciones con RA en contextos educativos. Esto permitió conformar la cadena de búsqueda con las combinaciones de las palabras: *"realidad aumentada" o "augmented reality"; "aplicaciones educativas"* o *"aplicación educativa"* o *"educational app"* o *"educational application*". Así, la cadena de búsqueda empleada fue: (*realidad aumentada OR augmented reality) AND (aplicaciones educativas OR aplicación educativa OR educational app OR educational application)*. Al ingresar esta cadena de búsqueda en cada una de las nueve fuentes indicadas al comienzo de esta sección, se recolectó un total de 2500 artículos.

c) Selección de artículos para su inclusión y exclusión. Para efectuar la selección de los artículos relevantes dentro del conjunto recolectado, se establecieron criterios de inclusión y de exclusión. Para este estudio no resultan relevantes: artículos duplicados, artículos no escritos en español o inglés, y artículos publicados con anterioridad al año 2015. Por otro lado, deben ser incluidos aquellos que, además de cumplir los criterios de inclusión mencionados, se vinculen específicamente con el desarrollo o uso de aplicaciones con RA en el campo de la educación, y que presenten un contenido con el adecuado nivel de detalle. A fin de detectar dicho nivel de detalle se efectuaron dos filtrados. Con el primer filtro se revisaron los títulos de las publicaciones y se leyeron sus resúmenes, detectando 52 artículos que verdaderamente podrían ser útiles para esta revisión. El segundo filtro consistió en la lectura completa de los artículos devueltos por el primer filtro, para evaluar en detalle los contenidos. Luego del segundo filtrado el conjunto de artículos se redujo a 23.

d) Codificación de resúmenes. Para codificar los resúmenes se decidió crear una tabla donde indicar para cada uno de los 23 artículos las siguientes características: nombre de la aplicación; país de origen; si se trata de una aplicación educativa o de una aplicación educativa colaborativa; software utilizado para su desarrollo; tipo de aplicación educativa según la clasificación propuesta por Yuen *et al.* [6];

dispositivos necesarios; nivel educativo para el que fue creada, existencia de experimentación con estudiantes reales; si se hizo análisis de resultados; área de estudio a la que corresponde; y finalmente, consideraciones vinculadas con Usabilidad y/o Interacción Persona-Ordenador (IPO). La tabla mencionada no se muestra por limitaciones de espacio.

e) Extracción de datos y mapeo de estudios. La extracción de información relevante de cada artículo seleccionado se realizó considerando las preguntas de investigación enunciadas en el apartado a) y la información que se obtuvo al ejecutar el apartado d), ambos descriptos en esta sección. Las respuestas elaboradas se enuncian en la próxima sección.

3. Análisis de la información recabada

Este mapeo sistemático permitió seleccionar 23 artículos en los que se identificaron 26 aplicaciones educativas que se valen de RA y que provienen de 13 países (Argentina, Reino Unido, Colombia, Indonesia, Malasia, España, China, Estados Unidos, Francia, México, Ecuador, Arabia Saudita y Perú). A continuación, se muestran las respuestas dadas a las 4 preguntas que guiaron este estudio.

Con respecto a la P1, se encontraron 26 aplicaciones educativas que utilizan RA, a saber: EnseñaApp, RatInARBox, ARTect, FLoNA, RutaDarwin, AROS, BioChemAR, ARabic-Kafa, ARPAC, MicroWorld, CRANIUM, Calculus, NumVo, Nupov, MolecularARWeb, Elements 4D, ARLoon, ARFlashcards, Anatomy 4D, Zookazam, SpaceCraft, y otras que no tienen nombre, a las que se identificó como App1, App2, App3 y App4.

Con respecto a la P2, se consideraron los tipos de aplicaciones educativas definidos por Yuen *et al*. (2011), es decir, ABD, MO, LI, ABJ y EH, y como puede verse en la Figura 1, la mayoría de las aplicaciones corresponden al tipo MO (60%). A este tipo corresponden RatINARBox, Solidos de Revolución, App1, ARTect, BiochemAR, App2, ARabic-Kafa, ARPAC, MicroWorld, CRANIUM, Calculus, NumVo, Nupov, App4, MoleculARWeb, Elements4D, ARLoon, ARFlashcards, Zookazam, SpaceCraft y Anatomy 4D. Un 17% pertenecen al tipo ABJ, es el caso de Enseña App, RutaDarwin, App3, CRANIUM, Calculus y Elements4D. Un 11% pertenecen al tipo LI, incluyendo a FLoNA, RutaDarwin, AROS y App3. Otro 12 % pertenece al tipo ABD, es el caso de Enseña App, ARPAC, MicroWorld y App4. Cabe aclarar que no se detectaron aplicaciones que respondieran al tipo EH.

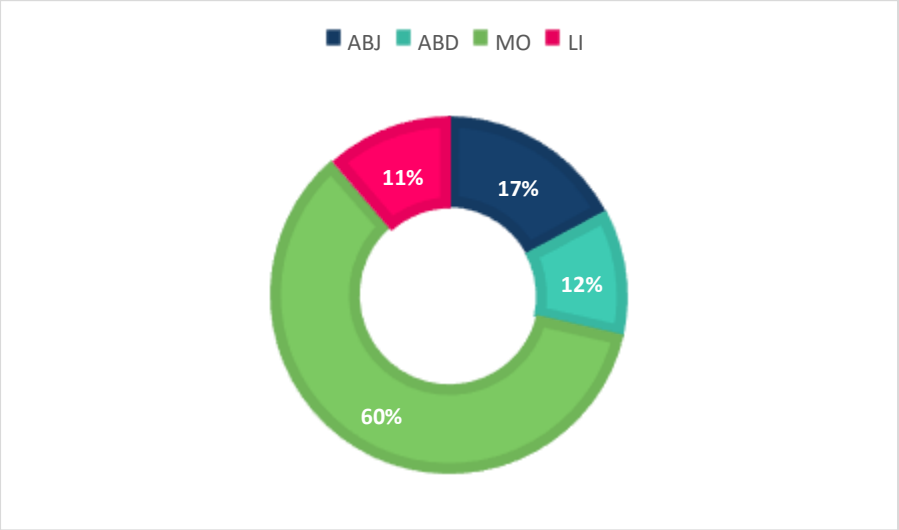


Fig. 1. Porcentajes de tipos de aplicaciones educativas que usan RA

Con respecto a la P3, se analizaron los siguientes aspectos para cada aplicación: el software que se utilizó para su desarrollo, el dispositivo que se necesita para su ejecución, el nivel educativo al que está dirigida, el área de estudio a la que pertenece, y si se ha experimentado su uso con alumnos reales y analizado los resultados obtenidos de dicha experimentación. Un 37% de las aplicaciones se desarrollaron utilizando Vuforia, otro 37% utilizó Unity, y un 3% se valió de HTML, CSS o Javascript. El 23% restante abarca los casos en que no se especificó el software de desarrollo. Al considerar los dispositivos necesarios para cada aplicación, se detectó que sólo una, MoleculARweb, podía funcionar tanto en computadora como en dispositivo móvil, el resto de las aplicaciones se creó para dispositivos móviles. En cuanto al nivel educativo de destino, sólo un 3% está orientado a nivel inicial. El resto de los niveles presenta porcentajes similares: 35% nivel primario, 30% secundario y 32% universitario. Al analizar el área de estudio al que se enfocan se reconoció una considerable variedad: Química, Biología, Inglés, Anatomía, Lengua, Geometría, Arquitectura, Informática. Sin embargo, tres de ellas presentan mayor frecuencia, Biología un 26%, Química un 16 % y Matemática un 13%. Teniendo en cuenta la experimentación concretada, se reconoció que sólo un 62% (16 aplicaciones) registran pruebas con estudiantes reales. De ellas, el 27% (7) efectuó el análisis de resultados observando el impacto de la RA sobre la calidad de la enseñanza y del aprendizaje, mientras que sólo en el 19% (5) de las aplicaciones se analizaron cuestiones de usabilidad (presentando resultados satisfactorios).

Con respecto a la P4, se reconoció que el 100% de las aplicaciones analizadas no presenta características colaborativas, es decir, las aplicaciones no permiten que los usuarios compartan información, dudas, conocimientos, ideas u opiniones, ni posibilitan el trabajo en equipo.

4. Conclusiones

Este artículo documenta un proceso de mapeo sistemático de literatura efectuado con el propósito de conocer el estado actual del conocimiento en el área de las aplicaciones que incluyen RA en contextos educativos. El proceso efectuado consideró 85 artículos recuperados de 9 fuentes bibliográficas, que finalmente se redujeron a 23 aplicando criterios específicos de inclusión y de exclusión, y que dio como resultado la identificación de 26 aplicaciones para estudio.

El trabajo efectuado permite formular las siguientes afirmaciones: a) en el período comprendido entre los años 2015 y 2023, se han generado varias investigaciones vinculadas con la creación y uso de RA en varios países, b) la RA favorece los procesos de enseñanza y de aprendizaje desde el aprender haciendo, ya que el alumno puede visualizar de una nueva forma los conceptos que a diario ve de forma escrita y puede experimentar con ellos de manera visual para entender mejor los conceptos o reforzar sus conocimientos, c) existen varias investigaciones concretadas en el campo de las aplicaciones educativas que cuentan con recursos aumentados, las que fijan su atención en la enseñanza de temas específicos de diversas áreas de estudio y en distintos niveles de enseñanza, d) La experimentación con estudiantes reales y el análisis de los resultados evaluando el impacto provocado en los aprendizajes no está siempre incluido en las investigaciones, e) los criterios de usabilidad de IPO reciben poca atención en el diseño y desarrollo de aplicaciones educativas que incluyan recursos aumentados, y finalmente, e) se detecta la inexistencia de aplicaciones educativas colaborativas que utilicen RA.

Corresponde destacar algunas afirmaciones, vinculadas con el impacto de la RA en contextos educativos, planteadas por los investigadores involucrados en los artículos analizados. Castellano y Santacruz [9] afirman que la RA favorece los procesos de enseñanza puesto que el profesor puede utilizar la tecnología de una forma creativa para favorecer su labor docente. Al mismo tiempo, los autores sostienen que la RA beneficia el proceso de aprendizaje del estudiante por tratarse de experiencias interactivas que requieren una actitud activa por parte del alumno, generando un incremento en su motivación mayor compromiso con su aprendizaje y permitiendo retener mayor cantidad de información. Sánchez Sordo [10] afirma que la RA es una alternativa adecuada para la enseñanza de contenidos prácticos, pues permite crear un escenario experimental donde el estudiante interactúa de modo directo con los objetos casi como lo haría en el aula tradicional. El autor sostiene que la RA impacta positiva y significativamente en la adquisición de conocimientos de los estudiantes. GómezVargas *et al.* [11] destacan que mediante aplicaciones con RA los estudiantes pueden practicar un mismo concepto hasta comprenderlo por completo. Kasinathan *et al.* [12] afirman que las aplicaciones con RA proporcionan un ambiente de aprendizaje relajado donde no hay limitaciones temporales, con recursos interactivos que despiertan el interés de los estudiantes.

Siendo la RA una tecnología emergente que está al alcance de todos, pues gran parte de la población tiene como mínimo acceso a un smartphone, es de esperar que en el futuro próximo se desarrollen nuevas aplicaciones educativas que incluyan RA. Seguramente, con el avance de la tecnología, también se expanda el abanico de dispositivos y periféricos incluidos en las nuevas aplicaciones.

Los resultados obtenidos mediante este mapeo sistemático de literatura demuestran la necesidad de desarrollar en el futuro aplicaciones web educativas y colaborativas que cuenten con recursos aumentados. Usando estas aplicaciones los estudiantes, organizados en pequeños grupos, se beneficiarán tanto de las ventajas que aporta el ACSC a sus procesos de aprendizaje, como de las reconocidas para la RA en contextos educativos.

Agradecimientos. Este trabajo fue financiado por la Universidad Nacional de Santiago del Estero (UNSE) a través del proyecto de investigación 23/C176-A-2022 "Desarrollo de aplicaciones para colaboración en e-learning" (2022-2025).

Referencias

1. Mendoza Morán, V. del R., Rivera Guevara, R., & Barriga Andrade, J.: Sistemas De Aprendizaje Colaborativo Móvil Con Realidad Aumentada. Revista Politécnica, 38(1), 67 (2016).
2. Pulido, R.: Estado del arte: Realidad aumentada con fines educativos. Escuela Coombiana de Carreras Industriales. ECCI Vol. 2 No. 3 (2013).
3. Azuma, R.: A survey of augmented reality. In Presence: Teleoperators and Virtual Environments 6, 4 (August 1997), 355-385 (1997).
4. Rigueros, C.: La realidad aumentada: lo que debemos conocer. TIA, 5(2), pp. 257-261 (2017).
5. Costaguta, R., Menini, M. & Gerez Martínez, L.: Analyzing some Experiences of Augmented Reality in Higher Education. Proceeding VIII Iberoamerican Conference of Human Computer Interaction HCI 2022. Editores: Vanessa Agredo-Delgado, Pablo H. Ruiz and Omar Correa. Editorial: CEUR Workshop. ISSN 1613-0073 (2023).
6. Yuen, S.C, Yaoyuneyong, G. & Johnson, E.: Augmented Reality: An Overview and Five Directions for AR in Education. Journal of Educational Technology Development and Exchange (JETDE): Vol. 4: Iss. 1, Article 11 (2011).
7. Costaguta, R., Menini M., Missio, D., Santana-Mansilla, P., Lescano, G., Martínez-Mirón, E.A. & Sánchez-Román, G. Aprendizaje colaborativo soportado por computadora. En: Perspectivas en la Interacción Humano-Tecnología. Jaime Muñoz-Arteaga, César A. Collazos,Toni Granollers, Huizilopoztli Luna-García (Compiladores). Editorial: HCICollab & Câmara Brasileira de Livro, SP Brasil, sección 5, capítulo 3 (2022).
8. Petersen, K., Feldt, R., Mujtaba, S., & Mattsson, M.: "Systematic mapping studies in software engineering". Proceedings of the 12th international conference on evaluation and assessment in software engineering, 68–77. Ciudad Real, Spain: ACM: British Computer Society (2008).
9. Castellano, T., Santacruz, L.: EnseñAPP: Aplicación Educativa de Realidad Aumentada para el Primer ciclo de Educación Primaria. Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología (2018).
10. Sánchez-Sordo, J. M. & Teodoro-Vite, S.: Desarrollo de un entorno de realidad aumentada para la enseñanza del condicionamiento operante en Psicología. Tecnología, Ciencia y Educación, 23, 115-136 (2022).
11. Gomez-Vargas, I., Medel-Esquivel, R., Garcia-Salcedo, R.: Realidad Aumentada como herramienta didáctica en geometría 3D. Lat. Am. J. Phys. Educ. Vol. 4, No.12 (2018).
12. Kasinathan, V., Al-Sharafi, A. T. A., Zamnah, A., Appadurai, N. K., Thiruchelvam, V., & Mustapha, A.: Augmented reality in ocean's secrets: educational application with attached book for students. Linguistics and Culture Review, 5(S1), 1123-1137 (2021).

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Una revisión sobre enfoques de agrupamiento automático en base a la personalidad y los estilos de aprendizaje de los estudiantes

César Ponce-Padilla[1], Germán Lescano[1,2] y Rosanna Costaguta[1],

[1] Instituto de Investigación en Informática y Sistemas de Información (IIISI), Universidad Nacional de Santiago del Estero (UNSE), Av. Belgrano (s) 1912, Santiago del Estero, Argentina

[2] Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)
ponce.padilla.c@gmail.com, {gelescano, rosanna}@unse.edu.ar

Resumen. El Aprendizaje Colaborativo Soportado por Computadora, en adelante ACSC, es una situación de aprendizaje mediada por computadora que busca promover el aprendizaje a través de la participación y el compromiso de los alumnos en procesos de colaboración. Sin embargo, la asignación de estudiantes en grupos no garantiza el éxito del proceso de aprendizaje. Esto explica el interés por el estudio de técnicas que permitan agrupar estudiantes de manera automática con la finalidad de incidir positivamente en los procesos de aprendizaje. Si bien muchos factores pueden impactar en el proceso de aprendizaje colaborativo, los aspectos cognitivos y sociales son de particular importancia. Por esta razón, en este trabajo se propone una revisión bibliográfica sobre la formación automática de grupos teniendo en cuenta estilos de aprendizaje y de personalidad de los estudiantes, dos criterios íntimamente involucrados con lo cognitivo y lo social. Los resultados de la revisión muestran que los estudios en el dominio son escasos, lo que evidencia la necesidad de continuar investigando en esta área.

Palabras clave: Aprendizaje Colaborativo Soportado por Computadora, Formación Automática de Grupos, Estilos de Aprendizaje, Estilos de Personalidad.

1. Introducción

El ACSC se vale de situaciones de enseñanza y de aprendizaje organizadas a través de actividades colaborativas, mediadas por computadora, en las que los estudiantes interactúan organizados en grupos realizando tareas asignadas por el docente [1]. Se reconoce que organizar estudiantes en grupos para realizar actividades colaborativas no garantiza automáticamente el éxito del aprendizaje [2].

Costaguta et al. [3] proporcionan una visión integradora de factores teóricos y prácticos del ACSC, incluyendo la diferenciación entre aprendizajes colaborativo y cooperativo, los fundamentos epistemológicos, la noción de grupo de aprendizaje, el papel del tutor, los roles de equipo, las emociones y la evaluación. A su vez, en otro trabajo, Costaguta et al. [4] describen cuatro desarrollos informáticos realizados para beneficiar los procesos de enseñanza y de aprendizaje en contextos de trabajo grupal mediados por tecnologías.

Con esto se visualiza que para lograr el aprendizaje en los entornos de ACSC se requiere de una cierta planificación y organización. Asimismo, se concuerda con

Järvenoja et al. [5] en que existen muchos factores que pueden impactar en el proceso de aprendizaje debido a la propia dinámica social del grupo de aprendizaje. Dentro de este contexto, el agrupamiento inteligente de estudiantes ha surgido como una estrategia prometedora para crear grupos de aprendizaje que maximicen la colaboración y el rendimiento de los alumnos. En este sentido, Costaguta [6] proporciona una revisión de técnicas de inteligencia artificial para la formación de grupos colaborativos comúnmente empleadas hasta el año 2015.

En su estudio sobre grupos de aprendizaje colaborativos Dang et al. [7] reconocen la predominancia de interacciones de tipo cognitivas y de ejecución de tareas en grupos colaborativos, identificando al mismo tiempo que la mayoría de los desafíos son del tipo cognitivos y sociales. De acuerdo con los datos exhibidos por Dang et al. [7] resulta importante que los grupos formados para situaciones de aprendizaje colaborativo tengan capacidades para desempeñarse en los dominios cognitivos y sociales.

En el plano educativo, los estilos de aprendizaje de los estudiantes pueden explicar los comportamientos de estos en el dominio cognitivo. En tanto que, en el plano social, la interacción entre los miembros de un grupo se puede ver afectada por sus personalidades. De esta asociación, consideramos de interés realizar una revisión bibliográfica que nos permita identificar antecedentes que hayan propuesto técnicas de formación de grupos en el dominio del ACSC teniendo en cuenta estilos de aprendizaje y estilos de personalidad.

La importancia de esta revisión radica en su potencial para informar y guiar futuras investigaciones en el campo del ACSC y el agrupamiento inteligente de estudiantes. Al comprender las técnicas existentes y su aplicabilidad, se pueden identificar áreas de mejora y oportunidades para desarrollar sistemas más precisos y personalizados. Además, se espera que los resultados de esta revisión contribuyan a la mejora de los entornos de aprendizaje colaborativo, facilitando una experiencia más enriquecedora para los estudiantes.

Este trabajo se estructura como sigue. En la sección 2 se enuncian los objetivos de la investigación. En la sección 3 se describe el proceso concretado para la búsqueda y selección de antecedentes. En la sección 4 se presentan los estudios incluidos, excluidos y validados. En la sección 5 se muestran las respuestas elaboradas para las preguntas de investigación que orientaron el presente trabajo y una breve descripción de los estudios analizados. Finalmente, en la sección 6 se presentan las conclusiones y una propuesta de trabajo futuro.

2. Objetivos de este estudio

El propósito de esta revisión es determinar el estado actual de conocimiento y desarrollo de sistemas que apliquen técnicas provenientes de la Inteligencia Artificial y del Aprendizaje de máquina para generar grupos de estudiantes en entornos de ACSC en base a sus estilos de aprendizaje y/o de personalidad.

Para formalizar el proceso de revisión se siguió la metodología de mapeo sistemático propuesto por Pettersen et al. [8]. Para ello se ejecutaron los cinco pasos de esta metodología: (i) definición de las preguntas de investigación, (ii) búsqueda de

estudios primarios, (iii) selección de artículos para su inclusión y exclusión, (iv) codificación de resúmenes, y (v) extracción de datos y mapeo de estudios.

3. Proceso de Búsqueda y Selección

Cumpliendo el primer paso de la metodología, se definieron las preguntas de investigación que orientaron este trabajo: 1) ¿Cuál es la evolución de los artículos publicados sobre este tema?, 2) ¿Qué enfoques se utilizan cuando se agrupan estudiantes por estilo de aprendizaje y/o estilo de personalidad?, y 3) ¿Qué técnicas se utilizan para formar de manera automática grupos de aprendizaje basados en estilos de aprendizaje y/o de personalidad de los alumnos?

Dando cumplimiento al segundo paso metodológico, se llevó a cabo la búsqueda de artículos relacionados con el tema de investigación se definieron palabras clave y sinónimos vinculadas con ellas. Se definieron dos cadenas de búsqueda, una en español y la otra en inglés. Para la búsqueda en español se utilizó la siguiente expresión de búsqueda: "formación de grupos y estilos de aprendizaje y estilos de personalidad". En tanto para la búsqueda en inglés se utilizó la cadena: *"group formation and learning style and personality style"*. Estas cadenas direccionaron la búsqueda en las siguientes fuentes bibliográficas: *ACM Digital Library*, *IEEE Digital Library*, *Science Direct* y *Google Scholar*.

Respondiendo al tercer paso metodológico, y a fin de determinar los artículos que deben ser seleccionados, se definieron criterios de inclusión y exclusión. Como criterios de inclusión se estableció que los artículos deben estar publicados en revistas (journals) o congresos, deben estar escritos en español o inglés, estar relacionados con la temática de agrupamiento de estudiantes mediante los criterios de estilo de aprendizaje y/o de personalidad, y la fecha de publicación debe ser mayor al año 2015. Se excluyen artículos publicados en libros, artículos escritos en otros idiomas distintos al español o inglés, artículos que no estén vinculados con la temática bajo investigación, y artículos cuya fecha de publicación sea menor al año 2015.

Para evaluar la calidad de los estudios seleccionados, se elaboró una lista de chequeo compuesta por 4 preguntas que fueron contestadas por sí o por no. Las preguntas fueron las siguientes: 1) ¿Está claro el objetivo de investigación?, 2) ¿Tiene concordancia el título con el desarrollo y/o la conclusión?, 3) ¿Están claros los descubrimientos realizados?, y 4) ¿Es una fuente de investigación primaria? Si bien con estas preguntas se busca determinar la calidad de los artículos seleccionados, no se planificó realizar eliminación de artículos en base a dicha calidad ya que los artículos pueden plantear algún enfoque de interés con respecto al agrupamiento de estudiantes.

Respondiendo al cuarto paso metodológico, con la intención de concretar la codificación de cada artículo seleccionado, se extrajeron los siguientes datos: año de la publicación, autores, técnicas de inteligencia artificial, estilos de aprendizaje, estilos de personalidad, métricas de evaluación, y una breve descripción del contenido.

4. Estudios Incluídos, Excluídos y Validados

A través de las búsquedas realizadas se obtuvo un total de 1765 artículos, de los cuales 1464 fueron extraídos de ACM, 12 de Google Scholar, 184 de IEEE y 105 de Science Direct. Muchos de los artículos mostrados en los resultados de búsqueda no se ajustaron a la temática bajo investigación, posiblemente por la generalidad de los términos usados para la búsqueda.

Luego de aplicar los criterios de inclusión y exclusión, y la lista de chequeo, sobre los 1765 artículos recuperados, el total se redujo a 9 artículos (3 de ellos provenientes de Google Scholar, 3 de ACM y 3 de IEEE) que cumplen con al menos 3 criterios de los 4 establecidos en la lista de chequeo.

5. Resultados

En esta sección se presentan los resultados de la revisión realizada sobre los artículos seleccionados. Como se muestra en Tabla 1, los atributos considerados fueron: año de publicación, técnica empleada para realizar la formación de grupos y criterio que se tuvo en cuenta para formar los grupos. En la Tabla 1, hay estudios que no introducen técnicas que permiten el agrupamiento automático de estudiantes, sin embargo, describen investigaciones sobre el agrupamiento de estudiantes teniendo en cuenta estilos de aprendizaje y/o de personalidad. Estos estudios que no introducen técnicas son descriptos brevemente en la sección 5.2.

Tabla 1. Algunas características de los artículos seleccionados.

Artículo	Año	Técnica empleada para realizar la agrupación automática	Criterio empleado para formar grupos
[9]	2015	Programación lógica con restricciones	Felder & Silverman, Personalidad (Big Five)
[10]	2016	No aplica	Felder & Silverman
[11]	2019	Algoritmo genético	Felder & Silverman
[12]	2016	Algoritmo genético	Felder & Silverman
[13]	2021	Algoritmo genético	Personalidad (Big Five)
[14]	2016	No aplica	Personalidad (Big Five)
[15]	2016	No aplica	Personalidad (test DISC)
[16]	2017	No aplica	Felder & Silverman, Personalidad (Big Five)
[17]	2023	No aplica	Personalidad (Big Five)

Cumpliendo el quinto y último paso metodológico, se llevó a cabo la extracción de datos y mapeo de los artículos. La información obtenida se muestra en las secciones que continúan. Además, cabe destacar que con esto se dio respuesta a las tres preguntas de investigación formuladas inicialmente.

5.1 Distribución de artículos por años

En esta subsección se responde a la primera pregunta de investigación. De los artículos seleccionados, se observa una escasa investigación sobre el estudio de técnicas

para formar grupos de aprendizaje en entornos de ACSC que consideren el estilo de aprendizaje y/o la personalidad como criterios de agrupamiento. En Tabla 1 se observa que se tiene una publicación en el año 2015 [9], cuatro en el 2016 [10, 12, 14, 16], una en el 2017 [16], una en el 2019 [11], una en el 2021 [13] y una en el 2023 [17].

5.2 Enfoques empleados para agrupar estudiantes en base a sus estilos de aprendizaje y de personalidad

En la segunda pregunta de investigación estamos interesados en determinar los enfoques que se emplearon para agrupar estudiantes en base a estilos de aprendizaje y/o de personalidad. A partir de los artículos identificados, se observa que todos tuvieron en cuenta el modelo de estilos de aprendizaje de Felder & Silverman [18].

Felder & Silverman [18] sostienen que el estilo de aprendizaje resulta de la combinación de cuatro dimensiones: comprensión, percepción, entrada y procesamiento. Cada una de estas dimensiones puede tomar uno de dos valores posibles. Para la dimensión comprensión el valor resultante puede ser global o secuencial, para la dimensión percepción puede ser intuitivo o sensitivo, para la dimensión entrada verbal o visual, y finalmente, el valor para la dimensión procesamiento puede ser activo o reflexivo. Combinando estos valores posibles para las cuatro dimensiones resultan 16 estilos de aprendizaje diferentes.

Costaguta et al. [10] realizaron un estudio para demostrar la existencia de ciertas relaciones entre estilos de aprendizaje de los estudiantes en grupos colaborativos y el rendimiento académico de esos grupos. Para analizar las relaciones existentes utilizaron el algoritmo A Priori. Los autores indican que las reglas de asociación descubiertas por su estudio pueden servir como orientación para formar grupos efectivos teniendo en cuenta los estilos de aprendizaje.

Respecto a estilos de personalidad, en los artículos seleccionados que los consideran, mayoritariamente trabajaron con el modelo Big Five. El test de personalidad DiSC es tomado sólo por un artículo.

El modelo de personalidad Big Five, o Five-Factor Model (FFM), ha obtenido el mayor consenso en el área de la psicología y es uno de los más utilizados en la literatura [19, 20]. Las dimensiones de personalidad que contempla son Extraversión (E), Amabilidad (A), Escrupulosidad (C), Neuroticismo (N) y Apertura (O). Cada combinación de valores en las diferentes dimensiones genera un tipo de personalidad con diferente tendencia a comportarse, interactuar, reaccionar y razonar. Dado que el modelo Big Five es un modelo de personalidad puramente descriptivo, los psicólogos han desarrollado diversas pruebas y cuestionarios que evalúan cada uno de los cinco factores o dimensiones en los individuos.

Por su parte, el test DiSC [21] permite medir la personalidad y el estilo de comportamiento de una persona mediante la evaluación de cuatro dimensiones, todas igualmente importantes: la capacidad de dirección/conducción (D), la capacidad de influencia/inspiración (i), la estabilidad (S), y la correctitud/obediencia/grado de consciencia (C), y debido a estas dimensiones es que recibe su nombre [21].

A continuación se describen brevemente los trabajos más significativos.

Carro, R. & Sanchez Horreo, V. [16] realizaron un estudio para obtener criterios útiles tanto para la adaptación individual y la formación dinámica de grupos en sistemas

de aprendizaje colaborativos adaptativos. Para realizar el estudio analizaron características tales como el estilo de personalidad, la inteligencia y los estilos de aprendizaje. Respecto a la formación de grupos encontraron una tendencia a la formación de grupos homogéneos respecto a la forma de razonamiento, la extroversión, el nivel de consciencia, las habilidades verbales y la apertura a experiencias; y la formación heterogénea respecto a los estilos de personalidad de agradabilidad y neuroticismo. Para realizar los análisis respecto a los estilos de personalidad tuvieron en cuenta el modelo Big Five y con respecto a los estilos de aprendizaje consideraron el modelo de Felder & Silverman.

Mujkanovic, A. & Bollin, A. [14] realizaron un estudio para determinar en qué grado influye el proceso de formación de grupos en entornos de aprendizaje colaborativo. En el estudio se muestra que la formación basada en estilos de personalidad logró buenos resultados. Para el análisis de la personalidad de los estudiantes se tuvo en cuenta el modelo Big Five.

Kobayashi, A. et al. [17] analizaron cómo y cuándo las diferencias de personalidad entre un alumno y sus compañeros de grupo afectan la satisfacción del alumno. El estudio confirma que las diferencias de personalidad entre pares afectan significativamente el nivel de satisfacción del alumno. El estudio detalla que la satisfacción es afectada por a) el promedio de las diferencias de personalidad entre el alumno y cada compañero de grupo, lo cual refleja el grado con el cual el alumno es diferente en promedio que sus compañeros, y b) la diferencia de personalidad del compañero quien tiene la mayor similitud o diferencia de personalidad con respecto al alumno.

Lykourentzou et al. [15] muestran cómo la compatibilidad de la personalidad en los grupos afecta el rendimiento y la percepción individual. El estudio muestra que balancear las personalidades contribuye a un mejor rendimiento en las tareas colaborativas. Además, los equipos balanceados por personalidad exhiben menos conflictos y sus miembros reportan mayores niveles de satisfacción y aceptación. Para analizar la personalidad de los miembros de cada grupo usaron el test de personalidad DISC.

5.3 Técnicas para la formación de grupos en base a estilos de aprendizaje y de personalidad

Respecto a la tercera pregunta de investigación, las técnicas aplicadas para la formación automática de grupos son algoritmos genéticos y algoritmos basados en programación lógica restrictiva. A continuación, se describe brevemente a cada una de las técnicas utilizadas, y también se brinda una breve descripción de cada uno de los artículos analizados.

5.3.1 Algoritmos genéticos

Los algoritmos genéticos fueron inventados por John Holland en el año 1975. Holland propuso los algoritmos genéticos como un método heurístico basado en la supervivencia de los mejores. Estos tipos de algoritmos son adecuados para problemas de búsqueda y optimización [22]. Estos algoritmos crecieron en popularidad debido a

su capacidad para desenvolverse bien en muchos tipos diferentes de problemas [23]. Si bien no son algoritmos veloces, ofrecen buenas heurísticas para problemas combinatorios. Estos algoritmos parten de una población inicial de soluciones candidatas para un problema y evolucionan hacia una solución óptima basados en los principios de selección natural y supervivencia del más apto, aplicando operaciones de cruzamiento y mutación.

En la revisión realizada se reconocen dos antecedentes que aplicaron este tipo de técnica. Por un lado, Lescano et al. [12] proponen un algoritmo genético para la generación automática de grupos considerando los estilos de aprendizaje de sus miembros. La formación de grupos con algoritmos genéticos es un problema permutativo, por lo que se diseñaron operadores genéticos. Para la formación de los grupos se basaron en la investigación de Costaguta et al. [10] sobre formación de grupos basados en estilos de aprendizaje en las cuales se estudiaron ciertas reglas de asociación. Estas reglas de asociación fueron utilizadas como base en la función de fitness planteada en este trabajo. Esta línea de trabajo es continuada por Palavecino et al. [11] quienes proponen una herramienta software que crea grupos de estudiantes teniendo en cuenta sus estilos de aprendizaje, combinándolos de tal forma que se incrementen o maximicen sus rendimientos académicos. En dicho artículo se describen los objetivos de la investigación, el enfoque propuesto para concretar la formación automática de grupos colaborativos, y algunos antecedentes relevantes.

Por otro lado, Sánchez et al. [13] proponen un enfoque basado en algoritmos genéticos para formar grupos en escenarios de aprendizaje colaborativo, considerando los rasgos de personalidad de los estudiantes como criterio de agrupación. Esta formación se lleva a cabo en dos etapas: en la primera, se recolecta la información de los estudiantes a partir de un instrumento psicométrico basado en el modelo de personalidad Big Five; mientras que, en el segundo, esta información alimenta un algoritmo genético que se encarga de realizar la agrupación de forma iterativa, buscando una formación óptima. En el trabajo se realizaron experimentaciones para la formación de grupos homogéneos y heterogéneos con respecto a este criterio de personalidad.

5.3.2 Algoritmos basados en programación lógica restringida

La programación lógica restringida es una extensión de la programación lógica donde se permiten restricciones en el cuerpo de las cláusulas. El paradigma de la programación lógica se basa en una idea de programación declarativa única donde los programas no se componen de declaraciones (como en la programación imperativa) ni de funciones (como en la programación funcional), sino de implicaciones lógicas entre colecciones de predicados [24]

Tacadao, G. & Toledo, R. (2015) proponen un modelo basado en programación lógica con restricciones para la formación de grupos que satisfagan múltiples criterios. Estos criterios pueden ser establecidos por el docente (por ejemplo, estilos de aprendizaje, competencia individual, estilos de personalidad, entre otros) así como intereses de los propios alumnos. El modelo fue evaluado teniendo en cuenta la cantidad de restricciones que se satisfacen por cada grupo.

6. Conclusiones y Trabajo Futuro

En este trabajo se llevó a cabo una revisión sobre la formación automática de grupos teniendo en cuenta estilos de aprendizaje y/o de personalidad. La revisión fue realizada teniendo en cuenta la metodología planteada por Petersen et al [8] y se definió como período temporal las publicaciones efectuadas a partir del año 2015, (principalmente para complementar revisiones sobre formación de grupos registradas previo a ese año). Sin embargo, es posible afirmar que a la fecha existe poca investigación sobre la formación de grupos de manera automática utilizando de manera puntual los criterios de estilos de aprendizaje y/o de personalidad.

A partir de los artículos analizados, es posible reconocer que las técnicas utilizadas para la formación automática de grupos son los algoritmos genéticos y la programación lógica restrictiva. En el caso particular de agrupamiento por estilos de aprendizaje, fue el modelo de Felder & Silverman el utilizado por todos los autores. Es importante destacar que sólo se encontró un artículo relacionado con los estilos de personalidad en el ámbito de estudio, y que sus autores eligieron aplicar el modelo conocido como Big Five. Esto refleja que es un tema que aún no ha sido explorado en profundidad. Asimismo, tampoco se observa la existencia de estudios que hayan propuesto opciones de agrupamiento de estudiantes que combinen los estilos de aprendizaje con los de personalidad.

Dado lo expuesto, se plantea a futuro diseñar e implementar una herramienta basada en técnicas de inteligencia artificial que permita formar grupos de estudiantes colaborativos teniendo en cuenta de manera simultánea tanto sus estilos de aprendizaje como los de personalidad.

Referencias

1. Costaguta, R., Menini, M., Missio, D., Lescano, G. & Santana-Mansilla, P. (2021): Desarrollos informáticos para grupos de aprendizaje en contextos de colaboración on-line. In: Martinez, M., Juliá, M., del Pilar Domínguez, A., Rizo Patrón, M., Gulotta, M., Sarmiento, M., Benac, M., Ledesma, A. (eds.) La investigación científica en la universidad y sus aportes a la sociedad, pp. 1466-1506. EDUNSE, Argentina.
2. Soller, A. (2001): Supporting Social Interaction in an Intelligent Collaborative Learning System. International Journal of Artificial Intelligence in Education, 12(1), pp. 40–62.
3. Costaguta, R., Menini, M., Missio, D., Santana-Mansilla, P., Lescano, G., Martínez-Mirón, E. & Sánchez-Román, G. (2022): Aprendizaje Colaborativo Soportado por Computadora. En: Perspectivas en la Interacción HumanoTecnología. Jaime Muñoz-Arteaga, César A. Collazos,Toni Granollers, Huizilopoztli Luna-García (Compiladores). Ed.: HCI-Collab & Câmara Brasileira de Livro, SP Brasil, 2022, sección 5, capítulo 3. ISBN 978-65-00-49304-7.
4. Costaguta, R., Menini, M., Missio, D., Lescano, G. & Santana-Mansilla, P. (2021): Desarrollos informáticos para grupos de aprendizaje en contextos de colaboración on-line. En La investigación científica en la Universidad Nacional de Santiago del Estero y sus aportes a la sociedad, (Eds.) Martínez, M.; Juliá, M.; del Pilar Domínguez, A.; Gulotta, M.; Sarmiento, M.; Benac, M.; Ledesma, A. pp 14661506.

5. Järvenoja, H., Volet, S., & Järvelä, S. (2013): Regulation of emotions in socially challenging learning situations: an instrument to measure the adaptive and social nature of the regulation process. Educational Psychology, 33(1), pp. 31–58.
6. Costaguta, R. (2015): Algorithms and machine learning techniques in collaborative group formation. In: Pichardo Lagunas, O., Herrera Alcántara, O., Arroyo Figueroa, G. (eds.) MICAI 2015. LNCS, vol. 9414, pp. 249-258.
7. Dang, B., Nguyen, A., Hong, Y., Thi Nguyen, B. & Dang Tran, B. (2023): Revealing the Hidden Structure of Affective States During Emotion Regulation in Synchronous Online Collaborative Learning. Proceedings of the 56th Hawaii International Conference on System Sciences, pp. 14-23
8. Petersen, K., Feldt, R., Mujtaba, S., & Mattsson, M. (2008): Systematic mapping studies in software engineering. Proceedings of the 12th international conference on evaluation and assessment in software engineering, pp. 68–77. Ciudad Real, Spain: ACM: British Computer Society.
9. Tacadao, G. & Toledo, R. (2015): A Generic Model for the Group Formation Problem Using Constraint Logic Programming*. In: 2015 IEEE 15th International Conference on Advanced Learning Technologies, pp. 307-308, IEEE Press, Hualien, Taiwan.
10. Costaguta, R., Menini, M. & Lescano, G. (2016): Applying Data Mining to Discover Successful Collaborative Groups. Proc. ACM EATIS.
11. Palavecino, A., Lescano, G. & Costaguta, R. (2019): Formación Automática de Grupos Colaborativos Considerando Estilos de Aprendizaje y Rendimiento Académico, pp. 1–5.
12. Lescano, G., Costaguta, R. & Amandi, A. (2016): Genetic algorithm for automatic group formation considering student's learning styles. 8th Euro American Conference on Telematics and Information Systems (EATIS), Cartagena, Colombia, pp. 1-8.
13. Sánchez, O. R., Collazos, C. A., & Redondo, M. A. (2021). Automatic group organization for collaborative learning applying genetic algorithm techniques and the big five model. Mathematics, 9(13).
14. Mujkanovic, A. & Bollin, A. (2016): Improving Learning Outcomes Through Systematic Group Reformation. In: 2016 9th International Workshop on Cooperative and Human Aspects of Software Engineering, pp. 97-103, ACM Press, Austin, USA.
15. Lykourentzou, I., Antoniou, A., Naudet, Y. & Dow, S. (2016): Personality Matters: Balancing for Personality Types Leads to Better Outcomes for Crowd Teams. In: CSCW '16: Proceedings of the 19th ACM Conference on ComputerSupported Cooperative Work & Social Computing, pp. 260-273, ACM Press, New York.
16. Carro, R. & Sanchez Horreo, V. (2017): The Effect of Personality and Learning Styles on Individual and Collaborative Learning: Obtaining Criteria for Adaptation. In: 2015 IEEE 15th International Conference on Advanced Learning Technologies, pp. 1585-1590, IEEE Press, Atenas, Grecia.
17. Kobayashi, A., Ishikawa, Y., Ikeda, K., Kamisaka, D. & Legaspi, R. (2023): Composing Groups in Collaborative Learning by Pair Personality Differences. In: UMAP '23: Proceedings of the 31st ACM Conference on User Modeling, Adaptation and Personalization, pp. 116-123, ACM Press, New York.
18. Felder, R. & Silverman, L. (1988): Learning and Teaching Styles in Engineering Education Application," J. Eng. Educ., vol. 78, no. 7.
19. Aguilar, R., De Antonio, A. & Imbert, R. (2007): Searching Pancho's Soul: An Intelligent Virtual Agent for Human Teams. In Proceedings of the Electronics, Robotics and Automotive Mechanics Conference (CERMA 2007), IEEE: Morelos, Mexico, pp. 568–571.
20. Soto, C., Kronauer, A. & Liang, J. (2015): Five-Factor Model of Personality. In The Encyclopedia of Adulthood and Aging; Krauss Whitbourne, S., Ed.; JohnWiley & Sons, Inc.: Hoboken, NJ, USA, pp. 1–5. ISBN 9781118528921.
21. Beedu, G. (2021): A study on the effectiveness of DiSC personality test. Tesis de doctorado en administración de recursos humanos. Universidad Selinus, Italia.

22. Sivanandam, S. (2008): Deepa, S.: Introduction to Genetic Algorithms. Springer.
23. Akman, O. (2014): Introducing Evolutionary Computing in Regression Analysis. Journal of Computational Science Education. 5(1), pp. 23-27.
24. Rossi, F. (2003): Chapter 15 - Constraint Logic Programming. In: Dechter, R. (eds.) Constraint Processing. In The Morgan Kaufmann Series in Artificial Intelligence, pp. 413-440, Morgan Kaufmann.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Uso de videojuegos en contextos de aprendizaje colaborativo: Un mapeo sistemático de literatura

Guillermo González[1], Walter Martin Anriquez-Atia[1] y Rosanna Costaguta[1]

[1] Universidad Nacional de Santiago del Estero, Instituto de Investigación en Informática y Sistemas de Información, Avda. Belgrano (S) 1917, 4200 Santiago del Estero, Argentina
guillermogonzalez.edu@gmail.com, {matia, rosanna}@unse.edu.ar

Resumen. Este artículo presenta un mapeo sistemático de literatura vinculado con la creación y uso de videojuegos en contextos de aprendizaje colaborativo soportado por computadora. Las fuentes de exploración consideradas fueron: Science Direct, Google Académico, ACM Digital Library, IEEE Digital Library, Scielo y Scopus. La búsqueda se realizó considerando las producciones publicadas en español o en inglés, entre los años 2015 y 2023. Inicialmente se obtuvieron 845 artículos que, luego de concretar las acciones propias de la metodología elegida, se redujeron a 22. Un detallado análisis de los artículos resultantes permite afirmar que el uso de los videojuegos provocó mejoras en el desempeño de los estudiantes, y que es necesario contar con un equipo interdisciplinario para la creación de videojuegos educativos colaborativos. A su vez, se identificaron las principales técnicas, herramientas y metodologías utilizadas hasta la fecha en los desarrollos.

Palabras clave: aprendizaje colaborativo soportado por computadora, videojuegos, desarrollo de videojuegos

1. Introducción

El aprendizaje colaborativo (AC) es un proceso social en el que, a partir del trabajo conjunto y el establecimiento de metas comunes, se genera una construcción de conocimientos; de acuerdo con Guitert y Giménez [24] se da cierta reciprocidad entre un conjunto de individuos que saben diferenciar y contrastar sus puntos de vista, de tal manera que llegan a forjar un proceso de construcción de conocimiento. Cuando esas experiencias de AC incluyen el uso de medios computacionales, se habla de Aprendizaje Colaborativo Soportado por Computadora (ACSC) [26].

El ACSC comparte las características esenciales del AC, pero incorpora los beneficios asociados con un soporte tecnológico, particularmente útil para este tipo de aprendizaje, puesto que hace posible que los participantes puedan contribuir desde cualquier lugar y en cualquier momento [23]. Las experiencias de ACSC apuntan entonces, a entender el aprendizaje como un proceso social de construcción de conocimiento colaborativo, en el que la computadora actúa como un mediador [27].

Actualmente los videojuegos están muy difundidos como medio de entretenimiento. Sin embargo, dado que son recursos que se enmarcan en el llamado Aprendizaje Basado en Juegos también resultan ser una valiosa herramienta pedagógica. Gros [25] sugiere denominarlos juegos serios (*serious games*) porque tienen como objetivo sacar provecho de las ventajas que proporcionan los videojuegos, pero cambian el foco desde el entretenimiento al aprendizaje. Así, los juegos serios aportan conocimientos mientras

se juega o se intenta dar solución a algún problema o reto planteado, combinando entretenimiento y aprendizaje. Mientras el jugador progresa en el juego, se va familiarizando con un tema específico y adquiriendo conocimientos acerca del mismo.

Con el propósito de investigar el uso dado a los videojuegos en el área del ACSC, se realizó el mapeo sistemático de literatura que se documenta en este artículo. En la siguiente sección se describen los pasos metodológicos seguidos, en la sección 3 se muestra el análisis efectuado sobre la información recolectada, y finalmente, en la sección 4 se enuncian algunas conclusiones.

2. Metodología

Los pasos metodológicos ejecutados fueron cinco, siguiendo la propuesta de Petersen et al. [28]: (i) definición de las preguntas de investigación, (ii) búsqueda de estudios primarios, (iii) selección de artículos para su inclusión y exclusión, (iv) codificación de resúmenes, y, (v) extracción de datos y mapeo de estudios.

i) Definición de las preguntas de investigación. Las preguntas se definieron a fin de recolectar información acerca del desarrollo y uso de videojuegos en los procesos de enseñanza y de aprendizaje en contextos de colaboración. Las preguntas formuladas fueron:
   - P1. ¿Qué lenguajes de programación se utilizaron para desarrollar videojuegos para educación en ACSC?
   - P2. ¿Cuál es el uso dado a los videojuegos en ACSC?
   - P3. ¿Cómo ha evolucionado el uso de videojuegos en ACSC?
   - P4. ¿Qué metodologías/técnicas se utilizaron para desarrollar los videojuegos en ACSC?
   - P5. ¿Qué requerimientos de hardware y software se satisfacen para desarrollar los videojuegos en ACSC?
   - P6. ¿Cuál fue el contexto educativo para el cual se desarrollaron los videojuegos en ACSC? (Nivel preescolar, primario, secundario, universitario)
   - P7. ¿Cuál fue la finalidad para la cual se desarrollaron los videojuegos en ACSC?
   - P8. ¿Para qué plataforma fueron desarrollados los videojuegos en ACSC? (hardware que necesita el usuario)

ii) Búsqueda de estudios primarios. La búsqueda de antecedentes se realizó de manera automática mediante los motores de búsqueda de las fuentes consultadas: Science Direct, Google Académico, ACM Digital Library, IEEE Digital Library, Scielo, y Scopus. Se tomaron como válidos artículos que trataran sobre el desarrollo y/o uso de videojuegos en el campo de la educación, tanto en inglés como en español. Las palabras clave de búsqueda fueron obtenidas a partir de un análisis de los conceptos relevantes para la creación y uso de videojuegos en contexto de ACSC. Esto permitió conformar la cadena de búsqueda con las combinaciones de las palabras:

"aprendizaje colaborativo", "aprendizaje colaborativo soportado por computadora", "ACSC", "collaborative learning", "computer supported collaborative learning" o "CSCL"; "videojuego", "video juego", "juegos serios", "videogame", "video game" o "serious game"; "desarrollo", "diseño", "uso", "development", "design" o "use". Así, la cadena de búsqueda empleada fue: "(aprendizaje colaborativo OR aprendizaje colaborativo soportado por computadora OR ACSC OR collaborative learning OR computer supported collaborative learning OR CSCL) AND (videojuego OR video juego OR juegos serios OR videogame OR video game OR serious game) AND (desarrollo OR diseño OR uso OR development OR design OR use)". Al ingresar esta cadena de búsqueda en cada una de las seis fuentes indicadas al comienzo de esta sección, se recolectó un total de 845 artículos.

iii) Selección de artículos para su inclusión y exclusión. Para efectuar la selección de los artículos relevantes dentro del conjunto recolectado, se establecieron criterios de inclusión/exclusión. Para este estudio no resultan relevantes: artículos duplicados, artículos no escritos en español o inglés, y artículos publicados con anterioridad al año 2010. Por otro lado, deben ser incluidos aquellos que, además de no verificar los criterios de exclusión mencionados, se vinculan específicamente con el desarrollo o uso de videojuegos en el campo de la educación, y presentan su contenido con el adecuado nivel de detalle. A fin de detectar dicho nivel de detalle se efectuaron tres filtrados. Con el primer filtro se eliminaron aquellos que estaban duplicados o que eran previos al 2010, quedando 683 artículos que verdaderamente podrían ser útiles para esta revisión. El segundo filtro consistió en la lectura de los títulos, resúmenes y conclusiones de los artículos devueltos por el primer filtro. Con este segundo filtrado el conjunto de artículos se redujo a 40. Finalmente, con el tercer filtrado se obtuvieron 22 artículos, los cuales cumplían con los criterios de inclusión y exclusión, que por la calidad de su contenido son relevantes para nuestro estudio.

iv) Codificación de resúmenes. Para codificar los resúmenes se decidió crear una tabla donde indicar para cada uno de los 22 artículos las siguientes características: título del artículo, autor/es, año de publicación, país de origen, si incluye desarrollo de videojuego, y resumen del artículo. La tabla mencionada no se muestra por limitaciones de espacio.

v) Extracción de datos y mapeo de estudios

La extracción de información relevante de cada artículo seleccionado se realizó considerando las preguntas de investigación enunciadas en el apartado i) y la información que se obtuvo al ejecutar el apartado iv) (ambos descriptos en esta sección). Las respuestas elaboradas se enuncian en la próxima sección.

3. Análisis de la información recabada

Este mapeo sistemático permitió seleccionar 22 artículos mediante los cuales se identificaron 21 videojuegos que soportan el ACSC. A continuación, se muestran las respuestas dadas a las preguntas (P1 a P8) que guiaron este estudio.

Con respecto a la P1, de los 22 artículos estudiados, sólo 8 [3, 4, 5, 6, 8, 11, 12, 14] mencionan el lenguaje de programación utilizado para el desarrollo de los videojuegos. 10 utilizaron Unity (combinándolo con los lenguajes de programación C, C++ y JavaScript), 1 utilizó Java, y 1 JavaScript.

Con respecto a la P2, los videojuegos fueron desarrollados para complementar los procesos de enseñanza y aprendizaje en el marco de una asignatura en particular, 2 de ellos complementan las clases tradicionales de Matemática [1, 16], 2 en Lengua [1, 17], 1 en Electrostática [2], 1 en Patrimonio Cultural [3], 1 en Lengua de Signos [4], 1 en Razonamiento Lógico [5], 1 en Física [6], 1 en Lengua Extranjera [7], 3 en Trabajo en Equipo [8, 11, 22], 2 en Biología [12, 18], 1 en Procedimientos Quirúrgicos [21], 2 en Ciencias de la Computación [9, 20] y finalmente, 2 en Prevención de Conflictos Escolares [14, 19].

Con respecto a la P3, al comienzo del período temporal objeto de estudio (20102023) se desarrollaban videojuegos en 2D y 3D casi en igual proporción. Sin embargo, dado el avance en la tecnología disponible, a partir de 2017 aparecen videojuegos desarrollados en 3D utilizando periféricos exclusivos para la inmersión y en 2019 incluyendo realidad aumentada.

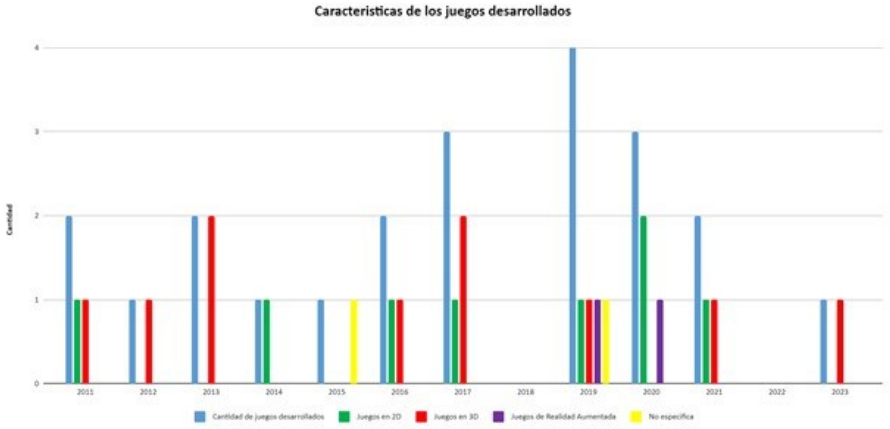


Figura 1. Características de los juegos desarrollados

Con respecto a la P4, de los 22 artículos analizados, sólo 10 mencionan la metodología de desarrollo seguida. Si bien no hay coincidencia de metodologías entre esas 10 menciones; se observó que comparten la participación de un equipo interdisciplinario durante las fases de diseño y desarrollo. La información se resume a continuación:

- En [2] la metodología usada es la de diseño y desarrollo de un juego educativo, que utiliza un marco de trabajo que consta de dos dimensiones: la dimensión educativa que incluye la identificación de los objetivos de aprendizaje y el modelo

pedagógico, y la dimensión lúdica donde se especifican los elementos del juego que aseguran una experiencia unificada y atractiva para los jugadores.

- En [4] se utiliza la metodología propuesta por Novak (2011) para el proceso de desarrollo del juego. La metodología consta de cuatro fases principales: concepto del juego, análisis y diseño, implementación y evaluación. Además, el proyecto se estructuró en tres etapas: desarrollo, experimento y evaluación. La primera etapa se centró en la creación del juego, la segunda en la recolección de datos sobre su uso en un entorno académico y la tercera en la evaluación de los resultados para identificar el impacto del juego en el proceso de aprendizaje. El equipo de desarrollo del juego estaba formado por miembros con experiencia en programación de juegos, diseño de juegos, usabilidad e interfaz de usuario, tecnología educativa, aprendizaje significativo y accesibilidad.
- En [5] la metodología utilizada fue la de diseño participativo (DP), en la que el público objetivo, en este caso, niños de primaria, toma un papel activo en el proceso de diseño del juego. En total, 10 niños de entre primero y quinto grado jugaron el juego y compartieron sus opiniones sobre sus limitaciones y sugerencias de mejora. Los investigadores utilizaron cuestionarios y sesiones de prueba grabadas para evaluar la facilidad de aprendizaje, el valor educativo y el compromiso del juego. Los resultados se utilizaron para realizar cambios en el juego. El juego se desarrolló en JavaScript con el motor de juegos Phaser.IO y está diseñado para ser jugado colaborativamente por cuatro niños en un entorno escolar, cada uno en su propia computadora.
- En [7] se utilizó una metodología que se basa en la definición de mecánicas de juego, la definición de objetivos de aprendizaje, la creación de desafíos progresivos que se adaptan a las habilidades del jugador y la posibilidad de colaboración entre los jugadores para fomentar el aprendizaje social. El enfoque general del diseño del juego es permitir que los jugadores construyan frases y conversaciones, lo que les da la oportunidad de practicar vocabulario, estructuras gramaticales y construir confianza en su capacidad para comunicarse en el idioma objetivo.
- En [8] la metodología utilizada incluye aspectos del diseño de juegos serios, el diseño de juegos multijugador y el aprendizaje colaborativo.
- En [9] la metodología utilizada para el diseño fue el enfoque de escenario, que se basa en el uso del lenguaje natural y un proceso de diseño iterativo. El enfoque de diseño centrado en el jugador también se utilizó en el diseño del juego. Las actividades de aprendizaje en el juego fueron diseñadas siguiendo los principios de aprendizaje colaborativo basado en juegos..
- En [10] la metodología utilizada en el proyecto incluye elementos adaptados de los procesos de desarrollo ágil de juegos y la investigación basada en el diseño (IBD). El enfoque de la IBD es ingenierizar formas particulares de aprendizaje y estudiarlas sistemáticamente. El equipo de diseño y desarrollo del juego utilizó el marco iterativo del desarrollo ágil para crear el prototipo inicial y continuar mejorándolo mediante pruebas de jugabilidad y diseño.
- En [12] la metodología utilizada para su desarrollo fue una combinación de enfoques de métodos mixtos, incluyendo sesiones de diseño participativo, entrevistas con estudiantes, pruebas de juego y ejercicios de diseño. La sesión de

diseño participativo involucró a 7 estudiantes de secundaria como codiseñadores del juego, mientras que las entrevistas semiestructuradas se realizaron con 15 estudiantes de secundaria. Las pruebas de juego se llevaron a cabo con 48 estudiantes de secundaria en un entorno de laboratorio y aula, quienes jugaron una versión beta del juego y proporcionaron comentarios sobre los aspectos que encontraron agradables o los aspectos que cambiarían. Se utilizó un enfoque de teoría fundamentada para codificar las respuestas de las entrevistas y las sesiones de prueba de juego. Finalmente, las opciones de diseño del juego se basaron en temas que surgieron durante el proceso iterativo de diseño.

- En [14] se utilizó una metodología interdisciplinaria que involucró a profesionales especializados en diferentes áreas del conocimiento, como la psicología, el diseño gráfico, la ingeniería computacional y la música. Estos expertos trabajaron juntos para transferir conocimientos disciplinares específicos hacia el desarrollo del producto de servicio que apunta a la transformación del tejido social de una comunidad.
- En [18] se utilizó una metodología que incluyó la definición de objetivos de aprendizaje, la identificación de los conceptos clave a enseñar, la creación de un guión gráfico y la implementación del juego en una plataforma móvil. Dicha metodología cuenta con las siguientes etapas: 1. Definición de objetivos de aprendizaje, 2. Identificación de conceptos clave que quieren enseñar a través del juego y diseño de actividades específicas para lograrlo, 3. Creación del guión gráfico del juego que incluye una historia, personajes, escenarios y situaciones a resolver, 4. Implementación del juego en una plataforma móvil, y fianlemente, 5. Evaluación del juego en relación con el desarrollo de habilidades de resolución de problemas y colaboración mediante la realización de encuestas y pruebas antes y después del uso del juego.
- En [1, 3, 6, 11, 13, 15, 16, 17, 19, 20, 21, 22] no se mencionan las metodologías utilizadas.

Con respecto a la P5, solo en [3] se menciona el hardware que se utilizó (procesador I7-4770k (8 CPU), tarjeta gráfica Nvidia GTX770, Memoria RAM de 16 gb, y periféricos de entrada Microsoft XBOX 360 Controller y Oculus Rift).

Con respecto a la P6, de los 22 artículos analizados: 8 desarrollaron videojuegos destinados para nivel primario [1, 4, 5, 11, 14, 15, 16, 17], 5 lo hicieron para nivel secundario [2, 4, 6, 12, 18], 4 para nivel universitario o superior [4, 9, 20, 21], y 5 no fueron creados para ser utilizados en un nivel educativo en particular [3, 7, 8, 10, 22].

Con respecto a la P7, todos los videojuegos desarrollados sirven como complemento de las técnicas tradicionales de enseñanza y de aprendizaje,y dan soporte al ACSC. Sin embargo, considerando finalidad, podemos dividir los artículos en dos grupos: el primero enfocado en el desarrollo de videojuegos multijugador para mejorar las habilidades de colaboración de los jugadores [1, 2, 7, 8, 9], y el segundo centrado en el desarrollo de videojuegos para personas con algún tipo de discapacidad (intelectual y motora en [22] y auditiva en [4]).

Con respecto a la P8, solo se pudo conocer la plataforma sobre la cual ejecutar los videojuegos en 14 artículos. Además, se observó que algunos videojuegos pueden ejecutarse en más de una plataforma. En la Tabla 1, se muestra esta información en

detalle, y también se indica si los videojuegos necesitan o no el uso de periféricos específicos.

Ref.	Tipo de Dispositivo	Periférico Necesario
[6, 11, 14, 16, 19]	Dispositivos móviles	Ninguno específico
[4, 12, 14, 21]	Computadores de escritorio	Ninguno específico
[5, 8, 16, 16, 19, 21]	Web	Ninguno específico
[1, 10]	Computadora	Pantalla táctil
[22]	Computadora	Proyector y Microsoft Kinect
[3]	Microsoft XBOX 360	Ninguno específico

Tabla 1. Plataformas y periféricos requeridos por los videojuegos analizados.

4. Conclusiones

Este artículo documenta el mapeo sistemático de literatura efectuado con el propósito de conocer el estado actual del conocimiento en el área de los videojuegos desarrollados en contextos educativos de ACSC. El proceso efectuado consideró 845 artículos recuperados desde seis fuentes bibliográficas, que finalmente se redujeron a 22 aplicando criterios específicos de inclusión/exclusión.

El estudio realizado permitió dar respuesta a las 8 preguntas de investigación inicialmente formuladas, lo cual llevó a identificar las diferentes técnicas, herramientas y metodologías utilizadas para el desarrollo de videojuegos en contextos de ACSC. Asimismo, permite formular las siguientes afirmaciones: a) en el período comprendido entre los años 2010 y 2023 se han generado varias investigaciones vinculadas con el desarrollo de videojuegos educativos colaborativos; b) los videojuegos complementan de manera efectiva los procesos de enseñanza y de aprendizaje en múltiples disciplinas, existiendo evidencias de que el uso de los videojuegos provocó mejoras en el desempeño de los estudiantes, generando en ellos una mejor predisposición para aprender y favoreciendo la colaboración con sus pares, c) para realizar un diseño, desarrollo y uso adecuado de videojuegos en contextos de ACSC se requiere contar con un equipo interdisciplinario (programadores, profesores, psicopedagogos, diseñadores gráficos, entre otros), y finalmente, d) los ejemplos de desarrollo y uso de videojuegos particularmente en educación superior y/o universitaria son escasos.

Dados los beneficios que aportan los videojuegos a los contextos colaborativos educativos, es de esperar que en el futuro su desarrollo y utilización aumente. Además, los resultados obtenidos mediante este mapeo sistemático de literatura demuestran la necesidad de desarrollar en el futuro aplicaciones web educativas y colaborativas que cuenten con recursos aumentados. Usando estas aplicaciones los estudiantes, organizados en pequeños grupos, se beneficiarán tanto de las ventajas que aporta el ACSC a sus procesos de aprendizaje, como de las reconocidas para la RA en contextos educativos.

Agradecimientos. Este trabajo fue financiado por la Universidad Nacional de Santiago del Estero (UNSE) a través del proyecto de investigación 23/C176-A-2022 "Desarrollo de aplicaciones para colaboración en e-learning" (2022-2025).

References

1. Craig, P. & Roa-Seïler, N. and Díaz, M & Rosano, F. (2014). A cognitonics approach to computer supported learning in the Mexican state of Oaxaca, 241-248.
2. Echeverría, A., García-Campo, C., Nussbaum, M., Gil, F., Villalta, M., Améstica, M. & Echeverría, S. (2011). A framework for the design and integration of collaborative classroom games. Computers & Education, 57(1), 1127–1136.
3. Andreoli, R., Corolla, A., Faggiano, A., Malandrino, D., Pirozzi, D., Ranaldi, M., Santangelo G. & Scarano, V. (2017). A Framework to Design, Develop, and Evaluate Immersive and Collaborative Serious Games in Cultural Heritage. Journal on Computing and Cultural Heritage, 11(1), 1–22.
4. Pontes, H., Furlan Duarte, J. & Pinheiro, P. (2018). An educational game to teach numbers in Brazilian Sign Language while having fun. Computers in Human Behavior.
5. Loparev, A., Sullivan, A., Verish, C., Westendorf, L., Davis, J., Flemings, M., Bers M. & Shaer, O. (2017). BacToMars. Proceedings of the International Conference on the Foundations of Digital Games - FDG '17.
6. Córdova Martínez M. & Alfonte Zapana, R. (2020). Collaborative game model for teaching physics using smartphone sensors. The 4th International Conference on Education and E-Learning.
7. Culbertson, G., Andersen, E., White, W., Zhang, D. & Jung, M. (2016). Crystallize: An Immersive, Collaborative Game for Second Language Learning. Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing - CSCW '16.
8. Wendel, V., Gutjahr, M., Göbel, S. & Steinmetz, R. (2013). Designing collaborative multiplayer serious games. Education and Information Technologies, 18(2), 287–308.
9. González-González, C. & Blanco-Izquierdo, F. (2012). Designing social videogames for educational uses. Computers & Education, 58(1), 250–262.
10. DeVane, B. M., Dietmeier, J., Missall, K., Nanda, S., Cox, M., Miller, B. J., Valentine E. & Dunkhase, D. M. (2019). Dropping into game design. Information and Learning Sciences, 120(9/10), 663–678.
11. López-Faican, L. & Jaen, J. (2020). EmoFindAR: Evaluation of a mobile multiplayer augmented reality game for primary school children. Computers & Education, 103814.
12. Perry, D., Aragon, C., Cruz, S., Peters, M. A. & Chowning, J. T. (2013). Human centered game design for bioinformatics and cyberinfrastructure learning. Proceedings of the Conference on Extreme Science and Engineering Discovery Environment Gateway to Discovery - XSEDE '13.
13. Martínez Borreguero G., Cortés T., Mateos M. & Correa F. (2023). Implicaciones cognitivas y emocionales de la implementación de un videojuego para el aprendizaje de contenidos de ciencias en Primaria. Revista Eureka sobre Enseñanza y Divulgación de las Ciencias. 20.
14. Esparza-Cervantes N., Guerrero Triana M., Cruz A., Mayor-Molinares Sr., Hoyos De Los Ríos O. & Restrepo D. (2020). La interdisciplinariedad en la construcción de Eru, un videojuego educativo. Arte, Individuo y Sociedad. 33. 71-85. 10.5209/aris.67028.
15. Tan, J. L., Goh, D. H.-L., Ang, R. P. & Huan, V. S. (2016). Learning efficacy and user acceptance of a game-based social skills learning environment. International Journal of Child-Computer Interaction, 9-10, 1–19.

16. Almar van der Stappen Y.L., Jiangxue X., Xiaoyu Y., Jingya L. & Erik D. van der Spek. 2019. MathBuilder: A Collaborative AR Math Game for Elementary School Students. In Extended Abstracts of the Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts (CHI PLAY '19 Extended Abstracts). Association for Computing Machinery, New York, NY, USA, 731–738.
17. Campos, I. (2021). Not too young to learn about the news: Best formats to educate about journalism in digital platforms. International Journal of Child-Computer Interaction, 27, 100218.
18. Sánchez, J. & Olivares, R. (2011). Problem solving and collaboration using mobile serious games. Computers & Education, 57(3), 1943–1952.
19. Schmitz, B., Schuffelen, P., Kreijns, K., Klemke, R. & Specht, M. (2015). Putting yourself in someone else's shoes: The impact of a location-based, collaborative roleplaying game on behaviour. Computers & Education, 85, 160–169.
20. Shi J., Shah A, Hedman G. & O'Rourke E. (2019). Pyrus: Designing A Collaborative Programming Game to Promote Problem Solving Behaviors. CHI '19: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems. 1-12.
21. Blanco A., Torrente J., Fernández-Manjón B., Ruiz P. & Giner M. (2017). Using a videogame to facilitate nursing and medical students first visit to the operating theatre. A randomized controlled trial. Nurse Education Today. 55.
22. Contreras M., Santos G. & García Bauza C. (2019). Videogame-based tool for learning in the motor, cognitive and socio-emotional domains for children with Intellectual Disability. Entertainment Computing. 30. 100301.
23. Costaguta R., Menini M. & Gerez Martínez (2023). Analyzing some Experiences of Augmented Reality in Higher Education. Proceeding VIII Iberoamerican Conference of Human Computer Interaction HCI 2022. Agredo-Delgado, V., Ruiz, P. & Correa, O. (Editores). Editorial: CEUR Workshop. ISSN 1613-0073.
24. Guitert, M. & Giménez, F. (2000). El trabajo cooperativo en entornos virtuales de aprendizaje. En J. Duart y A. Sangra (eds.). Aprender en la virtualidad. Barcelona: Gedisa, pp. 113-134.
25. Gros, B. (2000). El ordenador invisible. Barcelona: Gedisa.
26. Costaguta, R., Menini, M., Missio, D., Santana-Mansilla, P., Lescano, G., Martínez-Mirón, E. & Sánchez-Román, G. (2022). Aprendizaje Colaborativo Soportado por Computadora. En: Perspectivas en la Interacción Humano-Tecnología.
27. Muñoz-Arteaga, J., Collazos, C., Granollers, T. & Luna-García, H. (Compiladores). Editorial: HCI-Collab & Câmara Brasileira de Livro, SP Brasil, 2022, sección 5, capítulo 3. Disponible en https://hci-collab.uxartetic.com/activos/ libros/todos-loslibros.
28. Zañartu Correa, L. (2003). Aprendizaje colaborativo: una nueva forma de Diálogo Interpersonal y en Red. Revista Digital de Educación y Nuevas Tecnologías – Contexto Educativo, 28.
29. Petersen, K., Feldt, R., Mujtaba, S., & Mattsson, M. (2008). Systematic mapping studies in software engineering. Proceedings of the 12th international conference on evaluation and assessment in software engineering. Ciudad Real, Spain: ACM: British Computer Society, pp. 68-77.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

EVOTEM: una herramienta de software para el análisis de la evolución temática de publicaciones científicas en español

Santiago Bianco[1], Laura Lanzarini[2], and Alejandra Zangara[2]

[1] Grupo de Investigación en Sistemas de Información UNLa (GISI-UNLa)
[2] Instituto de Investigación en Informática LIDI (UNLP-CIC)
sabianco@unla.edu.ar, {laural,azangara}@lidi.info.unlp.edu.ar

Resumen A medida que transcurre el tiempo, los temas de estudio experimentan cambios significativos. Algunos alcanzan un mayor nivel de profundidad, dando paso a nuevos subtemas, mientras que otros se integran con temas existentes o quedan en el olvido. Estos cambios impactan en la terminología y complican el procesamiento de documentos de un tema específico provenientes de diferentes épocas. En el presente artículo se describe el desarrollo de una herramienta de software capaz de analizar la evolución temática de publicaciones en español siguiendo los pasos de la metodología sugerida en [1] y [2]. Esta herramienta fue sometida a dos juicios de expertos: el primero para verificar la veracidad de los resultados generados por la herramienta y el segundo para medir su facilidad de uso. En base a las respuestas obtenidas se puede afirmar que el software desarrollado es de utilidad para la disciplina y puede ser usado sin mayores complicaciones por usuarios no expertos.

Palabras Clave: Análisis Bibliométrico, Minería de Texto, Evolución Temática

1. Introducción

El análisis de documentos de textos provenientes de contextos específicos es un tema de interés para distintas áreas tales como recuperación de información, clasificación de documentos, análisis bibliométricos y scienciométricos, entre otros.

Cuando se procesan documentos de textos escritos en períodos de tiempo diferentes, la evolución temática es un aspecto que debe tenerse en cuenta. Reconocer los cambios, que se han ido produciendo a lo largo del tiempo, en la denominación de los distintos tópicos dentro a una misma disciplina o área de discurso, es una herramienta sumamente útil a la hora de querer aplicar estrategias pertenecientes a la Minería de Textos.

Como caso concreto, cualquier docente, investigador o estudiante que requiera hacer un artículo, tesis o trabajo de investigación, deberá efectuar una revisión del estado del arte correspondiente. En esta dirección, suele ser necesario identificar cuáles son los posibles temas de interés dentro de un dominio en particular. Este proceso de búsqueda bibliográfica generalmente consume mucho tiempo y si no se orienta correctamente puede conducir a bloqueos y frustración para el investigador. Sería interesante entonces contar con métodos y herramientas para simplificar la búsqueda y el análisis de bibliografía o publicaciones de cualquier tipo que faciliten estos procesos.

En primera instancia podrían usarse herramientas pertenecientes a la Bibliometría para comenzar a analizar textos de interés. La Bibliometría se conoce como una

disciplina capaz de describir un conjunto de publicaciones aplicando técnicas de análisis estadístico, identificando focos temáticos relevantes, redes de colaboración de autores, información sobre citas y demás. La Cienciometría es una subdisciplina de la Bibliometría que se enfoca particularmente en publicaciones científicas.

De todas formas, generalmente estos enfoques posibilitan análisis cuantitativos como por ejemplo una lista de autores más citados, instituciones que más publicaron, temas sobre los cuáles se escribió más, entre otros. En las ocasiones en que se requiera un análisis cualitativo más profundo deberán aplicarse técnicas de Minería de Textos y de visualización como mapas temáticos en conjunto con los métodos tradicionales de Bibliometría.

Los mapas temáticos son una forma de representar diferentes temas tratados en un campo de una disciplina científica en un determinado momento. Distinto tipo de información bibliométrica puede ser usada para armar estos gráficos, siendo una de ellas el análisis y la correlación entre términos relevantes [3].

De los mapas temáticos se deriva una técnica de análisis denominada evolución temática. La misma consiste en mostrar en una línea de tiempo la "evolución" de la relevancia de un tema en particular [3]. Por ejemplo, se podría mostrar que en el 2010 hubo un foco temático dedicado a la investigación en Redes Neuronales y el mismo grupo de gente que trabajaba en ese tema fue inclinando su investigación hacia otro distinto como puede ser la interpretabilidad de modelos de caja negra. La idea es mostrar que el primer tema mutó o evolucionó hacia el otro. Cabe destacar que en este contexto la palabra evolución denota cambio y transformación, y no necesariamente decir que "el Tema A evolucionó al Tema B" significa que el *Tema B* es mejor en algún aspecto con respecto al *Tema A*. [1].

Si bien existen algunas herramientas que permiten estos análisis, muy pocas se encuentran disponibles para el lenguaje español y no son tan intuitivas como para que puedan ser usadas por usuarios no expertos en el área informática. Además, están orientadas a publicaciones científicas en inglés descargadas de portales tales como Web of Science o Scopus, con un formato particular.

En este contexto, el presente artículo continúa el trabajo comenzado en [4], presentando la herramienta de software construida junto con los resultados de las evaluaciones realizadas por expertos.

2. Herramienta de software EVOTEM

Si bien existen algunas librerías o herramientas que permiten el análisis bibliométrico de textos [5], las mismas requieren tener conocimientos mínimos de estadística y minería de texto. Esto puede presentar dificultades para docentes, investigadores o posibles interesados en utilizarla que sean expertos en otras disciplinas. Por este motivo, la herramienta de software desarrollada busca no solo automatizar el análisis de la evolución temática sino también ser lo suficientemente intuitiva como para que pueda ser usada por usuarios no expertos sin mayores dificultades.

2.1 Metodología de desarrollo

El desarrollo de la herramienta está basado en una metodología prototipada evolutiva.

Una metodología prototipada permite hacer pequeñas iteraciones incrementales en el producto hasta generar el software final [6]. Esto no solo facilita su modificación luego de las pruebas de usabilidad y validación con expertos, sino que también permite la expansión de la herramienta a medida que avance la investigación, agregando nuevas métricas o procesos que mejoren los resultados.

Debido a los requisitos de usabilidad de la herramienta, la misma debe ser validada por usuarios para verificar que efectivamente no se necesitan conocimientos específicos para utilizarla. Además, se requiere que expertos validen los resultados finales que se obtengan a través de su uso. Es probable entonces que se necesiten modificaciones luego de dichas validaciones, por lo que se considera que la metodología elegida resulta adecuada.

2.2 Requisitos funcionales

La herramienta debe poder aplicar todos los pasos de la metodología detallada en [2] junto con las mejoras en la visualización allí propuestas. Puntualmente debe ser capaz de:

- Cargar artículos de revistas ya formateados para ser analizados por la herramienta.
- Procesar y filtrar automáticamente datos faltantes u outliers.
- Procesar y agregar los abstract de los artículos y las palabras clave a ser analizadas.
- Generar los gráficos correspondientes a la evolución temática de los textos a través de diagramas de Sankey.
- Permitir la visualización de los mapas temáticos generados para cada subperíodo analizado en la evolución temática.
- Filtrar el gráfico de la evolución temática de acuerdo al índice de inclusión de los ejes temáticos encontrados,

2.3 Requisitos no funcionales

Un requisito fundamental de la herramienta es que pueda ser utilizada por usuarios no expertos. Esto implica automatizar lo más posible la aplicación de la metodología y reducir al mínimo la parametrización de los algoritmos utilizados.

Una vez indicados los documentos a analizar, para visualizar el correspondiente diagrama que represente la evolución temática, el usuario sólo deberá indicar los años que dividen los subperíodos a analizar; el resto de los parámetros serán configurados automáticamente. Estos parámetros son:

- Cantidad máxima de términos a considerar en cada grupo temático.
- Frecuencia mínima de un término para ser considerado importante en un tema.
- Tamaño máximo de n-gramas a utilizar.

2.4 Implementación

El lenguaje elegido para desarrollar la herramienta es R, junto con el framework Shiny para la interfaz gráfica. Se opta por este lenguaje ya que permite implementar fácilmente varios de los algoritmos utilizados en la metodología. Además, cuenta con amplia documentación y, a través de la plataforma Shinny Apps, se puede distribuir fácilmente el software a través de la web. Esto último es fundamental ya que permite el acceso a la herramienta sin la necesidad de instalar nada por parte de los usuarios.

En la figura 1 se puede ver la interfaz de bienvenida de la herramienta. La forma de comenzar a trabajar es seleccionar alguno de los documentos de prueba ya procesados en el formato requerido que allí se indican o cargar datos propios. Una vez que se cargan los datos iniciales se genera automáticamente un análisis con parámetros por defecto, que pueden ser modificados posteriormente para explorar distintos subperíodos de tiempo, tal como se muestra en la figura 2. La herramienta permite ver tanto la evolución temática como los mapas temáticos de cada subperíodo establecido por el usuario. También en caso de ser necesario, permite ver detalladamente el conjunto de datos usados. En la figura 3 también puede verse la ayuda contextual que se incluye para facilitar la navegación de los usuarios que experimentan con EVOTEM por primera vez. [h]



Figura1. Pantalla inicial de EVOTEM



Figura2. Menú principal de EVOTEM

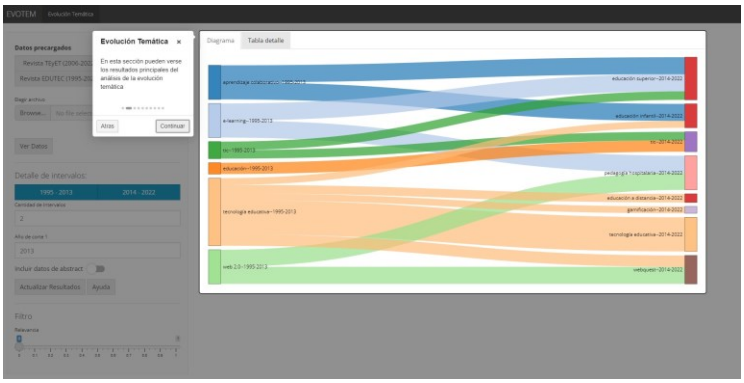


Figura3. Menú principal de EVOTEM con ayuda contextual

3. Juicio de expertos

Inicialmente, los resultados de la metodología llevada a cabo por la herramienta EVOTEM fueron validados por un único experto [2][4]. Sin embargo, para realizar una evaluación confiable se considera necesario ampliar este estudio realizando un juicio de expertos [7]. Con los resultados de dicho procedimiento se busca determinar si:

1. Los resultados proporcionados a los usuarios tienen sentido de acuerdo a la evolución del estado del arte en las áreas temáticas de las publicaciones analizadas.
2. Además de tener sentido, los resultados son interpretables y útiles para el usuario.
3. La navegación por los menués y su funcionalidad son suficientemente intuitivas como para poder ser comprendidas por cualquier investigador, docente o interesado sin que necesiten tener conocimientos en análisis de datos, estadística o similares.

3.1 Instrumentos

Para completar este juicio de expertos se construyen dos instrumentos con distintas preguntas, ambos implementados como formularios en Google Forms y enviados por mail a los expertos elegidos para responderlos. El primer instrumento permite evaluar los puntos (1) y (2) antes mencionados. La idea principal es mostrar a los expertos resultados otorgados por la herramienta para verificar su veracidad y legibilidad. En este caso, los datos usados para el análisis corresponden a publicaciones en revistas sobre tecnología en educación y educación en tecnología, por lo que se requieren expertos en estas áreas para responder al formulario. Dicho formulario se estructura de la siguiente manera:

Sección 1. Introducción con el objetivo del formulario, una breve descripción de la herramienta EVOTEM y del concepto de evolución temática. Se incluye además una imagen de ejemplo con indicaciones sobre qué representa cada elemento en el gráfico.

Sección 2. Preguntas sobre los datos personales del experto y experiencia en el área de tecnología en educación y educación en tecnología, las cuales se muestran en la tabla 1.

Sección 3. Evaluación de los resultados de la herramienta EVOTEM. En primer lugar se consulta al experto qué tan útil cree que sería una herramienta que proporcione una visión general de la evolución de las tendencias en un campo particular, dando a elegir como respuesta un número en la escala del uno al cinco, yendo de "Nada útil" a "Muy útil". Luego se presentan tres gráficos, dos de evolución temática y un mapa temático. En todos los casos se pregunta qué tan acertado cree que es el análisis que realiza la herramienta EVOTEM. Para cada uno también se puede responder en una escala del uno al cinco, siendo uno "No tiene nada que ver con lo que se investiga en el campo", y cinco "Representa casi perfectamente las tendencias de investigación en el campo".

El segundo instrumento corresponde a la evaluación de la usabilidad de la herramienta, junto con el acceso y la legibilidad de los resultados. Está conformado por las siguientes secciones:

Tabla 1. Preguntas sobre los antecedentes del experto.

Pregunta	**Tipo de respuesta**
Formación académica (Puede indicar título de grado y postgrado)	Texto corto libre
Área de experiencia profesional´	Texto corto libre
Institución donde trabaja	Texto corto libre
Cargo que desempeña en dicha Institución	Texto corto libre
¿Qué grado de especialización considera que posee respecto a la educación en tecnología y tecnología en educación?	Cerrada con las opciones: (1) No es un tema de gran interés para mí, prefiero otros; (2) Conozco de este tema, como de otros de mi especialidad; (3) Investigo y estudio el tema; y (4) Investigo y he escrito sobre el tema, con gran influencia en la comunidad científica

Sección 1. Introducción con el objetivo del formulario, instrucciones de uso de la herramienta y el link de acceso a la misma.

Sección 2. Nuevamente preguntas sobre los antecedentes del experto y su experiencia en el área, como se muestran en la tabla 1.

Sección 3. Evaluación de usabilidad de la herramienta. Se presentan cuatro preguntas con escalas numéricas del uno al cinco como se muestran en la tabla 2.

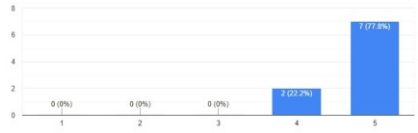
Además del instructivo en la Sección 1 del formulario, en el correo electrónico de distribución también se incluyen indicaciones de cómo probar la herramienta para evaluar los aspectos que se consideran relevantes y así orientar un poco la primer navegación de los expertos.

4. Resultados

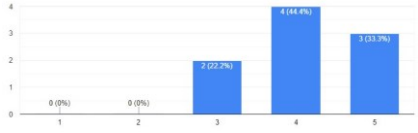
Los resultados obtenidos de la evaluación de expertos son muy positivos en cuanto a la utilidad de la herramienta EVOTEM, la calidad de resultados que entrega y la

usabilidad de la misma. Si bien no se cuenta con la respuesta de todos los expertos a los cuales se contactaron, el 75% respondió el formulario correspondiente al primer instrumento y el 60% respondió el segundo. Este último aún se encuentra en proceso, por lo que se espera disponer de nuevas respuestas a la brevedad. Todos los resultados del primer instrumento pueden observarse en la figura 4, mientras que los resultados parciales del segundo instrumento se encuentran en la figura 5.

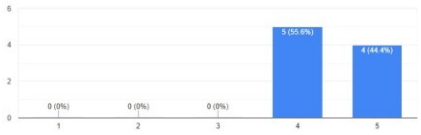
Como se observa en ambas figuras, los resultados son muy positivos para el primer instrumento y están siendo satisfactorios en el segundo, aunque este último aún está en proceso.



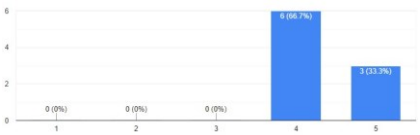
(a) A la hora de comenzar una investigación ¿qué tan útil cree que sería una herramienta que proporcione una visión general de la evolución de las tendencias en un campo particular?



(b) Teniendo en cuenta la evolución de las áreas temáticas representadas en la siguiente imagen, especifique qué tan acertado cree que es el análisis que realiza la herramienta EVOTEM

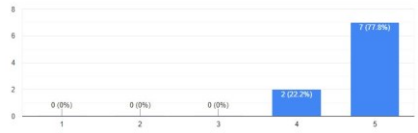


(c) A continuación se presentan otros resultados del análisis. Nuevamente especifique qué tan acertado cree que es el análisis de la herramienta EVOTEM

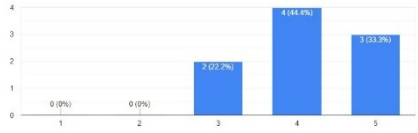


(d) Especifique que tan acertada es la siguiente clasificación de áreas temáticas obtenidas a partir del análisis hecho con la herramienta EVOTEM sobre la revista TEyET

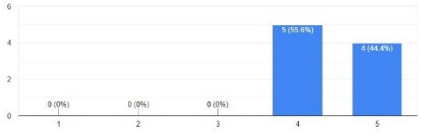
Figura 4. Resultados para las preguntas de evaluación de la herramienta del primer instrumento usado para la evaluación de expertos



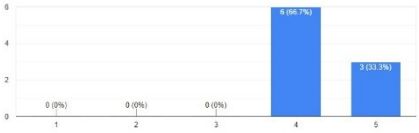
(a) ¿Qué tan claros le parecieron los resultados de evolución temática presentados por la herramienta?



(b) ¿Qué tan fácil de navegar le pareció la interfaz de la herramienta?



(c) ¿Qué tan sencillo le resultó probar distintos datos de prueba y parámetros para los resultados?



(d) ¿Qué tan útil le pareció la ayuda proporcionada por la herramienta?

Figura5. Resultados para las preguntas de evaluación de la usabilidad de la herramienta del correspondientes al segundo instrumento usado para la evaluación de expertos

Tabla 2. Preguntas del segundo instrumento sobre usabilidad de la herramienta

Pregunta	Grado inferior de la escala	Grado superior de la escala
¿Qué tan claros le parecieron los resultados de evolución temática presentados por la herramienta?	Muy poco claros	Muy claros
¿Qué tan fácil de navegar le pareció la interfaz de la herramienta?	Me costó encontrar las distintas opciones	Me resultó muy sencillo encontrar las opciones que buscaba
¿Qué tan sencillo le resultó probar distintos datos de prueba y parámetros para los resultados?	No pude cambiar los datos usados ni los parámetros	Pude visualizar distintos resultados con facilidad
¿Qué tan útil le pareció la ayuda proporcionada por la herramienta?	No me ayudó en nada	Muy útil

5. Conclusiones

En este artículo se presentó EVOTEM, una herramienta software desarrollada para facilitar el análisis de la evolución temática sobre publicaciones científicas en español por parte de usuarios no expertos en minería de texto y afines. Las primeras investigaciones realizadas por los mismos autores en este tema pueden consultarse en [1], [2] y [4].

La herramienta desarrollada cumple con dos objetivos específicos:

1. Es capaz de analizar artículos en español y entregar resultados consistentes, es decir, que estos resultados tengan sentido de acuerdo a la evolución del estado del arte de la disciplina original de los textos.
2. Su interfaz tiene es lo suficientemente simple como para poder ser usada y comprendida por cualquier tipo de usuario, sin que necesariamente esté familiarizado con los algoritmos de minería de texto y análisis de datos involucrados.

Con el objeto de corroborar el cumplimiento de ambos objetivos, se llevó adelante un juicio de expertos con dos instrumentos como se menciona en la sección 3.1.

Para aplicar el primer instrumento, se analizaron publicaciones en español de revistas de Tecnología en Educación y Educación en Tecnología. Posteriormente se le consultó a expertos en esta disciplina qué tan acertados les parecían los resultados del análisis. En base a sus respuestas (Figura 4) se puede concluir que:

- Se considera que una herramienta como EVOTEM es útil en el campo de la investigación.
- Los gráficos de evolución temática presentados por la herramienta condicen con la realidad.
- Los mapas temáticos de los cuáles se deriva la evolución temática también son acertados de acuerdo a la opinión de los expertos.

En el caso del segundo instrumento, si bien aún se están esperando algunas respuestas por parte de los expertos, de acuerdo a los datos que se tienen hasta el momento (Figura 5) se puede decir que la interfaz resulta lo suficientemente sencilla como para que pueda ser usada por cualquier tipo de usuario, aunque no cuenten con conocimientos específicos en minería de datos.

En base al juicio de expertos realizado se puede concluir entonces que EVOTEM cumple con los objetivos planteados al comienzo de la investigación [1], ya que proporciona resultados fiables y puede ser usado por usuarios no expertos.

6. Futuras líneas de trabajo

Si bien se considera que EVOTEM cumple con todos los objetivos planteados, aún hay mejoras que se pueden realizar.

En primer lugar, se puede mejorar la estabilidad y performance de la aplicación para que funcione más rápidamente. Esto puede realizarse mejorando los algoritmos de minería de texto usados, explorando otras librerías o con una breve refactorización del código.

Además, aún existen ciertos inconvenientes dependiendo de la forma en que los autores de las publicaciones escriben las palabras clave en sus artículos. Puede que dos palabras clave estén redactadas de tal forma que los algoritmos las consideren distintas, pero en realidad significan lo mismo. En estos casos, la herramienta las detecta como dos conceptos diferentes y puede que no se consigan los resultados más óptimos en el análisis. Podrían aplicarse algoritmos de lógica difusa y unificar todas estas palabras similares, pero un método tan directo puede no resultar apropiado para todos los escenarios posibles. Por ejemplo, "TIC" y "Tecnologías de la información y las comunicaciones" significan lo mismo a la hora del análisis, pero si se usa un algoritmo de comparación de cadena de caracteres con lógica difusa tradicional como la distancia de Levenshtein [8] daría como resultado que son conceptos completamente distintos. Sería de interés contar con un método fiable para detectar estas inconsistencias y mejorar aún más los resultados de la herramienta.

Referencias

1. S. Bianco, L. C. Lanzarini, and M. A. Zangara, "Evolución temática de publicaciones en español. una estrategia posible para el diseño de situaciones didácticas," in *XVI Congreso de Tecnología en Educación y Educación en Tecnología (TEyET 2021)*, RedUNCI, 2021.
2. S. Bianco, L. C. Lanzarini, and M. A. Zangara, "Thematic evolution of scientific publications in spanish," in *XXVII Congreso Argentino de Ciencias de la Computación (CACIC) (Modalidad virtual, 4 al 8 de octubre de 2021)*, 2021.
3. M. Cobo, A. López-Herrera, E. Herrera-Viedma, and F. Herrera, "An approach for detecting, quantifying, and visualizing the evolution of a research field: A practical application to the Fuzzy Sets Theory field," *Journal of Informetrics*, vol. 5, no. 1, pp. 146–166, 2011.
4. S. Bianco, L. Lanzarini, and A. Zangara, "Software tool for thematic evolution analysis of scientific publications in spanish," in *Argentine Congress of Computer Science*, pp. 64–77, Springer, 2021.
5. M. Aria and C. Cuccurullo, "bibliometrix: An r-tool for comprehensive science mapping analysis," *Journal of informetrics*, vol. 11, no. 4, pp. 959–975, 2017.
6. M. Carr and J. Verner, "Prototyping and software development approaches," *Department of Information Systems, City University of Hong Kong, Hong Kong*, pp. 319–338, 1997.

7. J. Escobar-Pérez and A. Cuervo-Martínez, "Validez de contenido y juicio de ex-´ pertos: una aproximación a su utilización," *Avances en medición*, vol. 6, no. 1, pp. 27–36, 2008.
8. N. Singla and D. Garg, "String matching algorithms and their applicability in various applications," *International journal of soft computing and engineering*, vol. 1, no. 6, pp. 218–222, 2012.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

INTEGRA+506: Una Metodología para el diseño y desarrollo de juegos serios

Juan Carlos Sandí-Delgado[1,2] [0000-0003-3932-3045] and Patricia Bazán[2] [0000-0001-6720-345X]

[1] Recinto de Guápiles - Sede del Atlántico, Universidad de Costa Rica, Costa Rica
juan.sandidelgado@ucr.ac.cr

[2] LINTI Facultad de Informática, Universidad Nacional de La Plata, Buenos Aires, Argentina
pbaz@mail.info.unlp.edu.ar

Resumen. La formación de competencias tecnológicas en educación superior se ha vuelto un factor imprescindible en el ámbito educativo. En este sentido, los juegos serios han sido utilizados para entrenamiento, la formación, capacitación y adquisición de habilidades en diferentes áreas del conocimiento. Por tal razón, se realizó una investigación con el objetivo de brindar una propuesta metodológica y arquitectónica que guíe paso a paso el diseño de juegos serios vinculados con la formación de dichas competencias. En consecuencia, se planteó una metodología y arquitectura de software para diseñar un prototipo no funcional del juego serio denominado SkillNet, el cual permite la adquisición de estas competencias en educación superior. Para llevar adelante la investigación, se realizó una indagación bajo un enfoque metodológico cualitativo y un diseño descriptivo, donde se realizó primeramente una revisión teórica acorde con los temas de investigación; principalmente los relacionados con metodologías y arquitecturas utilizadas para el diseño de juegos serios, así como las vinculadas con el diseño de prototipos. Posteriormente, se diseña y propone la metodología denominada INTEGRA+506 la cual se conforma de 6 etapas y quince pasos metodológicos que guían el diseño de juegos serios orientados a la formación de competencias tecnológicas. Por último, se concluye que la metodología INTEGRA+506 mostró efectividad para guiar el trabajo realizado en el diseño del prototipo no funcional de SkillNet y su aplicación ayudó a ordenar todo el proceso acorde con la validación realizada por el panel de expertos que revisó y analizó el prototipo.

Palabras clave: Juegos serios; diseño y desarrollo de juegos serios; desarrollo de competencias digitales; estrategias educativas; innovación educativa.

1. Introducción

Los juegos serios han sido utilizados para la formación en competencias digitales, entrenamiento y adquisición de habilidades en diferentes áreas del conocimiento, las cuales pueden contribuir a la generación de emociones y sensaciones, servir de apoyo para comprender diferentes procesos sociales, de salud, económicos y educativos (Khaleghi et al., 2022; Mercier & Lubart, 2023). En

esta línea, un juego serio se podría definir como aquel juego que tiene un enfoque que va más allá del entretenimiento o la diversión, el cual posee un propósito educativo explícito y cuidadosamente pensado, es decir, se puede utilizar para potenciar diferentes áreas del conocimiento, ofrecer un aprendizaje, apoyar procesos formativos, promover cambios de actitud o comportamiento, generar emociones, potenciar la adquisición de habilidades, entre otros. Caracterizado principalmente por ser atractivo, interactivo,

entretenido-dinámico, motivador, desafiante, fácil de utilizar, con la capacidad de estimular la colaboración, el razonamiento y el pensamiento crítico.

Actualmente, no se evidencia en la literatura consultada, diseñados o arquitecturas de estos juegos pensados exclusivamente para potenciar habilidades y competencias tecnológicas en el profesorado a nivel de educación superior. Por tal razón, resultó de interés elaborar una propuesta metodológica y arquitectónica que guíe paso a paso su diseño. En consecuencia, se planteó una metodología y arquitectura de software para diseñar un prototipo no funcional del juego serio denominado SkillNet, para permitir la formación de habilidades tecnológicas en el profesorado; específicamente las relacionadas con competencias y conocimientos básicos en redes de datos. La metodología se denominó INTEGRA+506 por la integración de metodologías de diseño y arquitectónicas (INTEGRA), luego, +506 por ser el código de Costa Rica, país de origen de la metodología.

Este tema es de alto interés, relevancia y trascendencia en la actualidad, debido a que aún existe una carencia en la definición de metodologías específicas para la creación y diseño de juegos serios en el ámbito del desarrollo de las competencias tecnológicas. Por ende, la metodología INTEGRA+506 es una contribución original dicha área de investigación, donde se realizó una investigación en profundidad en relación con todos los aspectos relacionados con procesos de diseño de juegos serios en el ámbito educativo. Se ha tenido en cuenta tanto el punto de vista educativo, como el metodológico y el tecnológico, incluyendo el estudio de las diferentes arquitecturas y los diferentes aspectos a considerar en el desarrollo de software.

2. Referente teórico

En esta sección se brinda una conceptualización de juegos serios y sus principales características, se adopta una definición propia, lo cual permite brindar contexto a la investigación realizada. Asimismo, se describe el concepto de competencias digitales.

2.1 Juegos Serios

El concepto de juegos serios se conoció por primera vez en 1970 por (Abt, 1970), quien indicó que los mismos tienen una intención educativa explicita, los cuales son cuidadosamente pensados y que su objetivo pedagógico va más allá del entretenimiento. Los juegos serios facilitan la adquisición de habilidades en diferentes áreas del conocimiento a través de la instrucción, simulación, entrenamiento o educación, facilitan principalmente el placer y la motivación (Abt, 1970; Kwon & Lee, 2016).

2.2 Competencias tecnológicas

El término competencias digitales ha sido asociado al dominio de variedad de herramientas tecnológicas, a las habilidades asociadas y requeridas para su correcta integración de estas en las diferentes áreas del conocimiento (Rodríguez-García et al., 2019).

En distintas investigaciones (Knezek et al., 2023; Sanz et al., 2022) se indica que las competencias digitales son consideradas como competencias claves, destacadas y transversales, debido a que potencian la formación de otras competencias, tales como la lingüística y la matemática, requeridas por la ciudadanía para garantizar una participación activa en la sociedad y la economía.

Al respecto, el Instituto Nacional de Tecnologías Educativas y de Formación del Profesorado (INTEF, 2017), señala que las competencias digitales están relacionadas con el uso creativo, seguro y crítico de las TIC para alcanzar los objetivos afines al entretenimiento, el aprendizaje, la inserción y contribución en la sociedad, apoyadas en las competencias básicas relacionadas al uso de tecnologías digitales. Además, dichas competencias están relacionas con el saber, saber hacer, saber estar y saber ser (Hernández et al., 2016; Lee et al., 2023).

Las competencias digitales se podrían definir como un conjunto de saberes relacionados con el correcto uso, dominio e integración de las tecnologías digitales y otras áreas del conocimiento relacionadas, que a través de diferentes habilidades, actitudes y valores permiten a las personas el saber ser, es decir, desarrollar un comportamiento propio ante un evento para ser capaz de potenciar el mejoramiento personal, académico o profesional, y con ello, garantizar su participación en la sociedad.

En la próxima sección, se aborda el marco metodológico seguido en la investigación, así como una descripción detallada de la metodología INTEGRA+506 (INTEGRA por la integración de metodologías de diseño y arquitectónicas y +506 por ser el código del país de origen de la metodología) la cual ha sido propuesta con el fin de integrar aspectos metodológicos y arquitectónicos en el diseño de juegos serios para potenciar en el profesorado la formación de competencias tecnológicas.

3. Metodología

Para alcanzar el objetivo propuesto, se realizó una investigación fundamentada en el diseño descriptivo, con un enfoque metodológico cualitativo (Hernández Sampieri et al., 2014), la cual se diseñó en tres fases; en la primera fase se realiza una búsqueda, revisión y selección de referencias bibliográficas relacionadas a los temas y objetivos propuestos en esta investigación. Lo cual permite consolidar el marco referencial de la investigación que incluye los temas de competencias digitales y juegos serios. En la segunda fase, se elabora la propuesta metodológica y arquitectónica (INTEGRA+506) que permite brindar solución al problema planteado (guiar paso a paso el proceso de planificación y desarrollo de Juegos Serios). Los pasos metodológicos propuestos, se formulan con base en las principales recomendaciones existentes en la literatura, identificación y análisis de metodologías y modelos propuestos para el diseño e implementación de video juegos y juegos serios que han sido comúnmente más utilizados, tales como EMERGO, EDoS, SAVIE, DODDEL, VGSCL, MECONESIS, MPIu+a y MPDSG, las cuales propiciaron el desarrollo del prototipo de juego serio denominado SkillNet.

En la tercera fase, se realiza la validación de la metodología propuesta a través de juicio de expertos y mediante su aplicación en el desarrollo de un prototipo de juego

serio denominado SkillNet, proceso que sienta las bases para la integración de metodologías y arquitecturas enfocadas en el diseño de juegos serios.

4. Propuesta metodológica y arquitectónica INTEGRA+506

En el presente apartado se aborda en detalle los pasos metodológicos y arquitectónicos presentes en la metodología INTEGRA+506 la cual guía paso a paso el diseño de juegos serios orientados a la formación de competencias tecnológicas, siguiendo las principales consideraciones teóricas en relación con el desarrollo de software, metodologías y arquitecturas para el diseño y desarrollo de juegos serios.

Por tanto, se propone una metodología compuesta por 6 fases, donde cada fase incluye diferentes pasos metodológicos y arquitectónicos (PM&A) que guía el diseño del prototipo no funcional del juego SkillNet. Para la definición de las fases de la metodología se consideraron aspectos teóricos del desarrollo de software, metodologías y arquitecturas para el desarrollo de videojuegos. Ahora bien, las fases como los pasos metodológicos podrían ser considerados de forma distinta dependiendo de la experiencia o área de formación de la persona o investigador que las analiza.

4.1 Fase 1. Consideraciones generales

Esta primera fase es la más importantes de la metodología, ya que se requiere definir los roles o actores participantes durante el desarrollo del prototipo, así como establecer las competencias a potenciar y el tipo de aplicación que se utilizará para ejecutarlo.

- **FM&A01 Definir roles o actores participantes -** El propósito de este paso consiste en identificar el equipo interdisciplinario, es decir, las personas expertas más recomendadas para participar en la definición de la propuesta del juego serio, tales como expertos pedagógicos, en el tema, de contenido, desarrolladores, pedagogos, entre otros (Spinelli & Massa, 2018), máxime que la definición de los requerimientos de usuario son influyentes en relación con la calidad de la pieza de software por desarrollar (Alshamrani & Bahattab, 2015).
- **FM&A02 Seleccionar competencias – Definir objetivos -** Al diseñar un juego serio para la formación de competencias tecnológicas, se debe considerar en primera instancia la definición de los requerimientos, luego, se define claramente los objetivos del juego, es decir, seleccionar previamente las competencias que se busca potenciar. En este sentido, existen estudios previos que han definido y clasificado las competencias en dimensiones e indicadores, tal es el caso de Prendes & Gutiérrez (2013) en España, Ministerio de Educación Nacional Colombiano (Sandí-Delgado, 2019) y por el Consejo Universitario de la Universidad de Costa Rica (UCR) (UCR, 2004). Lo anterior, coincide con lo indicado por (Yeh et al., 2017) quien agrega que para integrar las TIC en los procesos formativos es necesario poseer habilidades tecnológicas, pedagógicas y disciplinares, competencias que han sido también propuestas por el modelo de Conocimiento Técnico Pedagógico del Contenido (TPACK, *Technological Pedagogical Content Knowledge*).
- **FM&A03 Seleccionar el tipo de aplicación -** Posterior a la selección de las

competencias tecnológicas a potenciar en el profesorado a través del prototipo del juego serio por diseñar, se debe realizar la selección del tipo de aplicación sobre la cual se desarrollará el juego serio (Hainey et al., 2016), para ello, se debe considerar aquellos que hayan sido utilizados con mayor frecuencia para la ejecución de juegos serios. En este sentido, en una investigación realizada por (Sandí-Delgado et al., 2022), se consideran 5, entre ellas: para PC, las orientadas a trabajar en línea (web), video consolas, móviles y finalmente, las orientadas a la virtualidad. El tipo de aplicación a seleccionar va a depender de los objetivos propuestos y de los posibles recursos con que cuente la institución y el profesorado directamente involucrado. El prototipo de juego serio requiere de poder acceder a la aplicación desde cualquier computadora o dispositivo conectado a internet, el cual podría utilizar diferentes navegadores web (Albert & Torres, 2022a).

4.2 Fase 2. Lenguaje de programación

Para el diseño de una pieza de software y, en particular, el prototipo de un juego serio, se requiere de la definición de un paradigma y tipo de lenguaje de programación (IEEE Computer Society, 1990; Prasad & Verma, 2016).

- **FM&A04 Paradigma de programación -** Para el prototipo no funcional del juego serio, se requiere que se defina un paradigma de programación, ya que este indicará el método para su construcción, además, permitirá organizar las tareas que debe llevar a cabo la pieza de software, además, los mismos definen los estilos de programación (INET, 2016). Se podría elegir entre paradigmas de programación no estructurada, programación estructurada y programación orientada a objetos.
- **FM&A05 Tipo de lenguaje de programación -** En relación con el lenguaje de programación para crear la aplicación del prototipo no funcional, se puede utilizar una variedad de lenguajes enfocados en del desarrollo web. La estructura de una aplicación web consiste en el Frontend (la interfaz orientada al usuario), y el Backend (código interno para el manejo de datos del Frontend) acorde con lo indicado por (Gagliardi, 2021; Jones, 2020; Slivnik, 2022).
    1. Lenguajes para el desarrollo ***Frontend***: Comúnmente se utiliza para el desarrollo de *Frontend* lenguajes como JavaScript (brinda dinamismo a los sitios web), CSS (aporta estilos, diseños y formatos a la página web) y el Lenguaje de Marcado de Hipertexto (HTML, *HyperText Markup Language*) el cual provee una estructura de etiquetas y maquetado para el desarrollo del sitio web.
    2. Lenguajes para el desarrollo del ***Backend***: Existe variedad de lenguajes para desarrollo del *Backend*, entre ellos se encuentra PHP (*Hypertext Preprocessor*), Python, Java, C++, JavaScript, entre otros, los cuales pueden ser utilizados para el desarrollo del prototipo.

Dependerá del desarrollador cual lenguaje va a utilizar para el desarrollo del prototipo, máxime que la selección de este queda sujeta a la experiencia del usuario (desarrolladores) y de los objetivos por alcanzar.

4.3 Fase 3. Procesos de software

Acorde con lo indicado por (Sommerville, 2016), se debe definir un proceso de software para guiar todas las actividades relacionadas con la producción de la pieza de software, en este caso, el modelo de desarrollo y el diseño centrado en el usuario.

- **FM&A06 Modelo de desarrollo de software -** Para el diseño de la pieza de software por desarrollar, se podría utilizar un modelo de cascada, modelo de proceso incremental, modelo de prototipos, modelo espiral, modelo de desarrollo de aplicación rápida o bien, un modelo de ciclo de vida ágil.
- **FM&A07 Diseño centrado en el usuario -** Para el diseño del prototipo se requiere de un diseño centrado en el usuario para ofrecer una mayor satisfacción y experiencia a nivel del usuario (UX). Lo anterior, debido a que se requiere que los usuarios influyan sobre las decisiones de diseño y sobre los objetivos pretendidos por alcanzar con el prototipo. En relación con los estilos de componentes a nivel de interfaz para prototipos, se recomienda que para el manejo del diseño y presentación de la aplicación, se definan los estilos que presentará el sitio web, dentro del diseño el intérprete que asignará los estilos es el lenguaje Hojas de Estilo en Cascada (CSS *Cascading Style Sheets*), para la utilización de este, existe variedad de opciones a elegir, entre ellos *Frameworks* sencillos de utilizar, tales como Bootstrap, Bulma, MDBoostrap, que permiten brindar una identidad distinta al estilo, según el *Frameworks* elegido. De igual manera se podría utilizar CSS en su versión pura "vanilla", la cual se caracteriza por ser ligera, adaptable y de código abierto (Melgarejo-Torralba et al., 2022).

4.4 Fase 4. Ingeniería de requerimientos

Esta fase es de importancia en el desarrollo de piezas de software (Sommerville, 2016), las partes involucradas deben definir los requerimientos generales, ya que a como una buena especificación de requerimientos es determinante en la calidad del software (Gómez-Fuentes, 2011). Para ello, se debe considerar los requerimientos funcionales y no funcionales, así como definir claramente los casos de uso, las reglas de negocio y la ficha del juego.

- **FM&A08 Requerimientos funcionales y no funcionales -** Para el diseño de software se requiere definir claramente los servicios que el sistema debe proporcionar e indicar cuales serían las restricciones de esta.
- **FM&A09 Casos de uso -** Para el diseño del prototipo no funcional del juego serio, se requiere la descripción de las acciones o actividades que permitan concretar el proceso del diseño final (Sommerville, 2016).
- **FM&A10 Reglas de negocio -** Para el diseño del prototipo no funcional del juego serio, se requiere la definición de las reglas de negocios, las cuales tienen como objetivo definir restricciones o límites que permitan alcanzar el objetivo del proyecto.
- **FM&A11 Ficha de juego -** Los juegos serios usualmente, suelen tener asociada una ficha de juego, en la cual se cuente brevemente su **historia** (de qué trata el juego), así como ofrecer las **indicaciones del cómo jugarlo** y, finalmente, incluir el aspecto más caracterizante de los juegos serios, el cual corresponde a la

estrategia de ludificación, es decir, el juego describe la metodología de ludificación que utiliza (medallas, ranking, insignia, puntajes y premiación) (Sandí-Delgado & Sanz, 2020), lo cual es un aspecto relevante para considerar, debido a que los juegos serios se caracterizan por la intención pedagógica y, por incorporar estrategias lúdicas (Sandí-Delgado, 2019).

4.5 Fase 5. Diseño de software – Aspectos arquitectónicos

En el diseño de software se requiere generar una estructura que permita relatar el "como" se logra cada función propuesta del software mediante una especificación de propiedades de objeto (Leung & Cockburn, 2021). Entre las especificaciones, se requiere describir como mínimo el estilo arquitectónico (con todos sus componentes) y el diagrama de flujo de datos.

FM&A12 Estilo arquitectónico - El estilo arquitectónico puede estar integrado por diferentes componentes, los cuales varían acorde con la experiencia del desarrollador, el objetivo del juego y tipo de aplicación a la que está orientada la pieza de software.

1. **Modelo, Vista, Controlador** – En modelo, vista, controlador para prototipos, se requiere para separar de manera lógica la estructura de programación en una segregación por capas (López Caballero et al., 2019).
2. **Microservicios (web services)** – Se requiere la implementar un web services para el consumo de recursos almacenados en la base de datos donde estará alojada la información del juego, tales como información general del usuario y métricas: tiempo y puntuaciones (López Caballero et al., 2019).
3. **Entorno de desarrollo -** En relación con los estilos de componentes a nivel de interfaz para prototipos, se requiere seleccionar un Entorno de Desarrollo Integrado (IDE, *Integrated Development Environment*) adecuado al lenguaje que se va a utilizar (Bazán et al., 2017; Yulianto et al., 2017), en este sentido, las opciones a utilizar podrían ser amplias y con alta variedad de software entre el cual elegir. Dentro de las opciones más utilizadas en el mercado para el desarrollo web están: Visual Studio Code debido a su alta flexibilidad para trabajar como lenguaje, WebStorm y SublimeText, entre otros (Janssen et al., 2019; Yun et al., 2020).
4. **Framework o Librerías -** Para el desarrollo del prototipo no funcional del juego serio se podría utilizar un *Frameworks*, el cual permite el desarrollo a través de un estructura o esqueleto (es decir, es un marco de trabajo), el cual aporta flexibilidad para el manejo tanto del *Backend* como del *Frontend,* debido a que estos marcos de trabajo cuentan con librerías creadas por la misma comunidad de desarrolladores que los utilizan (Leung & Cockburn, 2021), estos podrían facilitar el desarrollo de piezas de software dependiendo de las circunstancias a implementar (Cabrera, 2017). En la actualidad, existe una variedad de Frameworks para JavaScript tales como React, AngularJS, Vue.js (Albert & Torres, 2022b), cada uno posee sus características particulares a nivel de usabilidad y robustez.
5. **Gestor de base de datos -** Se requiere utilizar un Sistema Gestor de Base de Datos (DBMS, *Database Management System*) para administrar el acceso por parte de los usuarios a la base de datos, así como permitir la realización de consultas y generación de informes en relación con los avances de los jugadores (Hueso-

Ibáñez, 2015). El DBMS usualmente cumple con el objetivo de servir de interfaz entre esta, el usuario y las aplicaciones (Díaz-Erazo et al., 2022). Para el almacenamiento de los datos se podría utilizar varios gestores, tales como MySQL, SQL, Oracle, entre otras. Estos gestores permiten almacenar ordenadamente la información generada por los jugadores.

6. **Alojamiento -** En el prototipo no funcional del juego serio se requiere alojar la aplicación en la web para garantizar su disponibilidad y alcance al usuario. Lo más utilizado actualmente, son la contratación de servicios de proveedores de computación en la nube (Cobeña, 2021). Algunas opciones para alojar la aplicación (juego, API, entre otros) son Azure, AWS, *Google Cloud Platform*, entre otras (Albert & Torres, 2022a), las cuales usualmente ofrecen un costo económico por el servicio consumido de alojamiento. Igualmente, se podría valorar la opción de buscar alternativas gratuitas como GitHub.
7. **Autentificación de usuarios -** En el prototipo no funcional del juego serio se requiere de la autentificación de usuarios, la cual se podría realizar de distintas maneras; utilizar librerías como (JWT, *Json Web Token*) o Auth0, las cuales brindan seguridad garantizada en el control de acceso de los usuarios (Díaz-Erazo et al., 2022). Este proceso es requerido para garantizar la privacidad, integridad y calidad de los datos de cada jugador, así como mantener un registro individual de la actividad y alcances de cada usuario.
8. **Versionado -** El prototipo de juego serio requiere ofrecer una aplicación simple, que se mantenga una única versión compartida para todos los clientes (Sandí-Delgado & Bazán, 2017). Se podrían utilizar herramientas existentes para el control de versiones, tales como Git (sistema de control de versiones distribuido), el cual ofrece diferentes clientes tales como GitHub y GitLab, Subversión (sistema de control de versiones centralizado) y Mercurial (sistema de control de versiones multiplataforma), siendo este último el menos utilizado (Albert & Torres, 2022a). La diferencia más importante entre Git y Subversión consiste en que, en los sistemas centralizados (Subversión) se mantiene únicamente una copia de la aplicación y las copias permanecen respaldadas en un solo lugar. En los sistemas distribuidos que hacen uso de Git, se mantiene una copia del proyecto original en cada equipo local y los cambios se actualizan o suben al repositorio en línea, los cuales posteriormente deben ser descargados de dicho repositorio para actualizar la versión de cada uno de los desarrolladores (Albert & Torres, 2022a). El versionado de los diferentes repositorios (*frontend, backend*) de las aplicaciones web se pueden actualizar sin interrumpir su flujo, al desplegar la aplicación con la nueva versión del repositorio la misma se actualiza con sus nuevas funcionalidades o correcciones.

FM&A13 Diagramas de clases (UML) y Diagrama entidad relación (ER) - En el diseño del prototipo se requiere de la definición de un diagrama de clases y un diagrama entidad relación debido a que este permite identificar claramente cuál será el flujo de la información durante el desarrollo de la pieza de software. Para ello, se pueden utilizar herramientas gratuitas tales como Draw.io, StarUML.io, Diagrams.net, o bien de paga como Visio. En estos diagramas se hace uso de simbología tal como el

círculo, diferentes tipos de flechas, rectángulos, texto entre otros, los cuales permiten visualizar la entrada y salida de datos (Sommerville, 2016).

4.6 Fase 6. Codificación del software

En el diseño del prototipo se sugiere contemplar la codificación del software, debido a que permite traducir todas las especificaciones diseñadas en código fuente, haciendo más fácil el leer y comprender la pieza de software desarrollada (Prasad & Verma, 2016). Lo anterior, permite obtener un estilo de programación, pautas de codificación, técnicas de codificación estructuradas y técnicas de verificación del código. En esta fase se analiza tanto las características del código de software como la reutilización de sus componentes (Sommerville, 2016).

- **FM&A14 Características del código de software - E**l código del software debe estar escrito en congruencia con los requisitos del usuario, es decir, apegado a los estándares y buenas prácticas de programación, eficiente en sus algoritmos, con manejo de excepciones, validaciones y medidas de seguridad, garantizando un mínimo margen de errores (Prasad & Verma, 2016). Se pueden implementar algunas de las siguientes medidas: simplicidad de código (simple y conciso), modularidad (dividido en varios módulos), diseño (estilo específico y consistente), eficiencia (uso óptimo de los recursos disponibles), claridad (simplicidad, legibilidad y modularidad), accesibilidad (fácilmente disponibles y accesibles) o estabilidad (funcionar correctamente en diferentes plataformas sin afectar su diseño y consistencia).
- **FM&A15 Reutilización de componentes -** En el diseño del prototipo se sugiere contemplar la reutilización de código y componentes dinámicos de software, máxime si se piensa en una aplicación web, ya que facilita el mantenimiento, escalabilidad y funcionalidad de la aplicación (Prasad & Verma, 2016). La reutilización del software es posible de implementar en diferentes niveles: Nivel de abstracción, nivel del objeto, nivel del componente y nivel del sistema (Sommerville, 2016).

En la Figura 1, se puede observar el resumen de las fases y pasos metodológicos y arquitectónicos de la metodología INTEGRA+506.

4.7 Validación de la propuesta a través de un prototipo

En el desarrollo de software se recomienda que las propuestas metodológicas sean validadas por profesionales y expertos en el área, por ello, en este caso, se recomienda diseñar un prototipo de juego serio siguiendo los pasos metodológicos propuestos a través de INTEGRA+506, posteriormente, hacer uso del juicio de expertos para validar el prototipo diseñado.

Fase	Pasos metodológicos y arquitectónicos	
Fase 1. Consideraciones generales	FM&A01 FM&A02 FM&A03	Definir roles o actores participantes. Seleccionar competencias – Definir objetivos. Seleccionar el tipo de aplicación.
Fase 2. Lenguaje de programación	FM&A04 FM&A05	Paradigma de programación. Tipo de lenguaje de programación.
Fase 3. Procesos de software	FM&A06 FM&A07	Modelo de desarrollo de software. Diseño centrado en el usuario.
Fase 4. Ingeniería de requerimientos	FM&A08 FM&A09 FM&A10 FM&A11	Requerimientos funcionales y no funcionales. Casos de uso. Reglas de negocio. Ficha de juego.
Fase 5. Diseño de software – aspectos arquitectónicos	FM&A12	Estilo arquitectónico: ✓ Modelo, vista, controlador. ✓ Microservicios (web service). ✓ Entornos de desarrollo. ✓ Framework o Librerías. ✓ Gestor de base de datos. ✓ Alojamiento. ✓ Autentificación de usuarios. ✓ Versionado.
	FM&A13	Diagramas de clases (UML) y Diagrama Entidad Relación (ER).
Fase 6. Codificación del software	FM&A14 FM&A15	Características del código de software. Reutilización de componentes

Fig. 1. Fases y pasos metodológicos y arquitectónicos de INTEGRA+506

5. Prototipo SkillNet

El prototipo con funcionalidad mínima SkillNet puede ser accedido desde el siguiente enlace: https://skillnetucr.github.io/Prototipe/ para iniciar sesión se debe utilizar en ambos casos como usuario y contraseña: admin123@gmail.com

Este juego serio tiene como propósito la formación de habilidades tecnológicas en el profesorado; específicamente las relacionadas con competencias y conocimientos básicos en redes de datos. El juego consta de tres etapas conformadas por diferentes niveles cada una, en las cuales el jugador debe enfrentar desafíos para superarlos y desbloquear el siguiente nivel. El juego tiene un personaje principal llamado Robotín.

Robotín es un dispositivo de red que siempre ha tenido un sueño, conocer la isla del Caribe (San Andrés - Colombia). Robotín es un equipo que se encuentra en un Centro de Datos en Limón (Costa Rica) y debe salir de su entorno para tomar una embarcación (WAN), eso sí, para salir no la tendrá tan fácil, debe superar una serie desafíos para poder pasar de su ubicación a la Isla.

El Centro de Datos está compuesto por salidas de seguridad, cada puerta podría abrirse en base a conocimientos en el área de Redes y Comunicaciones de Datos. Para poder cumplir con el sueño de Robotín se tiene que pasar por 3 ambientes o escenarios, los cuáles son los siguientes:

- Robotín está conectado en un MDF (*Main Distribution Frame*), para poder desconectarse y salir debe resolver desafíos sobre clases de direcciones del protocolo de internet, asignación de máscaras variables y subdivisión de redes, al

completar estos ejercicios puede salir al salón principal de tecnologías de información.

- En el salón principal, la clave de salida es el ordenamiento e identificación del cableado necesario para la creación de una red y los dispositivos intermediarios que la componen, donde una vez que Robotín consiga ordenar dicho salón, puede salir y embarcarse en su travesía en el Crucero Island San Andrés.
- Ya en el crucero, debe Robotín enseñarle al capitán la trayectoria del viaje a San Andrés, para completar este desafío debe conocer la confección correcta de una red del protocolo de enrutamiento. Cada vez que supera un reto y desbloquea un nivel (paso de una ubicación a otra), la dificultad aumenta y recibe como premio un alimento básico de sobrevivencia (agua, proteína, medicamentos, entre otros).

¿Cómo se gana el juego? – El jugador debe superar todas las actividades evaluativas de un nivel para lograr desbloquear el siguiente, en este caso se le abrirá una puerta que le permitirá seguir avanzando en el juego. Para ganar, debe desbloquear los 3 niveles y completar absolutamente todas las actividades en cada uno. **Estrategia de ludificación** - El prototipo ofrece 5 estrategias a saber: Sistema de puntuación, Desafíos, Niveles, Intentos por etapa y Sistema de ranking.

La metodología INTEGRA+506 cumplió con el objetivo de guiar el diseño del prototipo no funcional del juego serio llamado SkillNet, el cual está orientado a la formación de competencias tecnológicas en el profesorado.

6. Conclusiones

Se definió una metodología y arquitectura de software para el diseño de juegos serios para la formación de competencias tecnológicas. A partir de la metodología propuesta denominada INTEGRA+506 se diseñó un prototipo con funcionalidad mínima de un juego serio denominada SkillNet, para potenciar competencias digitales en el profesorado, específicamente vinculadas con conocimientos básicos en redes de datos.

En relación con las TIC y competencias tecnológicas, se caracterizaron ampliamente ambos términos, a partir de sus aspectos caracterizantes identificados, se brindaron definiciones propias, resultado de una investigación teórica realizada. Luego, se indagó el concepto de juegos serios y se identificaron juegos que se están utilizando para el desarrollo de competencias tecnológicas. Se brindó una definición propia. Se evidenció que existe una ausencia de metodologías guíen el diseño de juegos serios específicamente para la formación de competencias tecnológicas. Por ello, se analizaron propuestas de metodologías y arquitecturas para el diseño de videojuegos y juegos serios en particular, a partir de esto, se elaboró una propuesta metodológica y arquitectónica para el desarrollo de juegos serios orientados a la formación de competencias tecnológicas denominada INTEGRA+506, la cual mostró efectividad para guiar el diseño del prototipo SkillNet. Su aplicación ayudó a ordenar todo el proceso. Sin embargo, resta aún la parte funcional y ampliar su aplicación a otros escenarios educativos.

La metodología INTEGRA+506 podrían resultar de muy alta relevancia y trascendencia a nivel nacional e internacional para investigadores y desarrolladoras de videojuegos serios, ya que les podría permitir tomarla como guía, base o referente al momento de desarrollar juegos serios. Es importante indicar que la metodología en mención permitió desarrollar el prototipo de SkillNet, pero el mismo aún no ha sido implementado o aplicado con docentes, lo cual se espera realizar en un mediado plazo, en su debido momento, se describirá los resultados que se obtengan en relación con el uso del prototipo en espacios de formación en competencias tecnológicas. Se debe trabajar en la aplicación de la propuesta metodológica INTEGRA+506 para el diseño de prototipos de juegos serios en diferentes ámbitos educativos, así como en distintas áreas del conocimiento y competencias.

Referencias

1. Abt, C. C. (1970). *Serious Games*. Viking Press.
2. Albert, M., & Torres, M. V. (2022a). Desarrollo de una aplicación web como catálogo de buenas prácticas de accesibilidad. Universitat Politècnica de València.
3. Albert, M., & Torres, M. V. (2022b). *Desarrollo de una aplicación web como catálogo de buenas prácticas de accesibilidad* [Universitat Politècnica de València]. http://hdl.handle.net/10251/188224
4. Alshamrani, A., & Bahattab, A. (2015). A Comparison Between Three SDLC Models Waterfall Model , Spiral Model , and Incremental / Iterative Model. *International Journal of Computer Science*, *12*(1), 106–111. https://ijcsi.org/papers/IJCSI-12-1-1-106-111.pdf
5. Bazán, P. A., Fernández, A., Del Río, N., Molinari, L., Pérez, J. P., & Banchoff, M. (2017). *Aplicaciones, servicios y procesos distribuidos: Una visión para la construcción de software* (P. A. Bazán (ed.); 1st ed.). Editorial de la Universidad Nacional de La Plata (EDULP). http://hdl.handle.net/10915/62354
6. Cabrera, J. (2017). *Modular Design Frameworks: A Projects-based Guide for UI/UX Designers* (K. Jameson (ed.); 1st ed.). Apress. https://doi.org/10.1007/978-1-4842-1688-0
7. Cobeña, J. A. (2021). *Plataformas de computación en la nube para el desarrollo de aplicaciones web* [Pontificia Universidad Católica del Ecuador Sede Esmeraldas]. https://repositorio.pucese.edu.ec/handle/123456789/2595
8. Díaz-Erazo, A. D., Morales-Morales, R. M., Pineda-Chávez, V. K., & Morales-Cardoso, L. S. (2022). Comparative Analysis of performance for SQL and NoSQL Databases. *17th Iberian Conference on Information Systems and Technologies*, 1–14. https://doi.org/10.23919/CISTI54924.2022.9820292
9. Gagliardi, V. (2021). Decoupled Django: Understand and Build Decoupled Django Architectures for JavaScript Front-ends. In J. Markham (Ed.), *Decoupled Django*. Apress. https://doi.org/10.1007/978-14842-7144-5
10. Gómez-Fuentes, M. del C. (2011). *Notas Del Curso Análisis De Requerimientos* (M. del C. Gómez (ed.); 1st ed.). Universidad Autónoma Metropolitana, Unidad Cuajimalpa. http://www.cua.uam.mx/pdfs/conoce/libroselec/Notas_Analisis_Requerimiento.pdf
11. Hainey, T., Connolly, T. M., Boyle, E. A., Wilson, A., & Razak, A. (2016). A systematic literature review of games-based learning empirical evidence in primary education. *Computers & Education*, *102* (November), 202–223. https://doi.org/10.1016/j.compedu.2016.09.001
12. Hernández, C. A., Arévalo, M. A., & Gamboa, A. A. (2016). Competencias TIC para el desarrollo profesional docente en educación básica. *Praxis & Saber. Revista de Investigación y Pedagogía*, *7*(14), 41–69. https://doi.org/10.19053/22160159.5217

13. Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, M. del P. (2014). Análisis de datos cuantitativos. In *Metodología de la investigación* (6th ed.). Mc Graw Hill. https://www.uca.ac.cr/wpcontent/uploads/2017/10/Investigacion.pdf
14. Hueso-Ibáñez, L. (2015). Administración de Sistemas Gestores de Bases de Datos (2nd ed.). RA-Ma.
15. IEEE Computer Society. (1990). IEEE Std 610.12-1990, IEEE Standard Glossary of Software Engineering Terminology. *Ieee*, *121990*(1), 1. https://doi.org/10.1109/IEEESTD.1990.101064
16. INET. (2016). *Programación Orientada a Objetos*. http://ceaer.edu.ar/wp-content/uploads/2018/04/ApunteTeorico-de-Programacion-OO.pdf
17. INTEF. (2017). *Marco Común de Competencia Digital Docente*. Instituto Nacional de Tecnologías Educativas y de Formación del Profesorado (INTEF). http://educalab.es/documents/10180/12809/Marco+competencia+digital+docente+2017/afb079871ad6-4b2d-bdc8-58e9faeeccea
18. Janssen, J., Surendralal, S., Lysogorskiy, Y., Todorova, M., Hickel, T., Drautz, R., & Neugebauer, J. (2019). pyiron: An integrated development environment for computational materials science. *Computational Materials Science*, *163*, 24–36. https://doi.org/10.1016/j.commatsci.2018.07.043
19. Jones, C. B. (2020). *Understanding Programming Languages*. Springer Nature Switzerland. https://doi.org/10.1007/s11023-007-9062-6
20. Khaleghi, A., Aghaei, Z., & Behnamghader, M. (2022). Developing two game-based interventions for dyslexia therapeutic interventions using gamification and serious games approaches entertainment computing journal. *Entertainment Computing*, *42*, 100482. https://doi.org/10.1016/j.entcom.2022.100482
21. Knezek, G., Christensen, R., Smits, A., Tondeur, J., & Voogt, J. (2023). Strategies for developing digital competencies in teachers: Towards a multidimensional Synthesis of Qualitative Data (SQD) survey instrument. *Computers & Education*, *193*, 104674. https://doi.org/10.1016/j.compedu.2022.104674
22. Kwon, J., & Lee, Y. (2016). Serious games for the job training of persons with developmental disabilities. *Computers & Education*, *95*(April), 328–339. https://doi.org/10.1016/j.compedu.2016.02.001
23. Lee, G., Caton, E., & Ding, A. (2023). Evaluating digital competencies for pharmacists. *Research in Social and Administrative Pharmacy*, *19*(5), 753–757. https://doi.org/10.1016/j.sapharm.2023.01.012
24. Leung, J., & Cockburn, A. (2021). Design Framework for Interactive Highlighting Techniques. *Design Framework for Interactive Highlighting Techniques*, *14*(2–3), 1–176. https://doi.org/10.1561/9781680839258
25. López Caballero, V., Fragoso Díaz, O. G., Santaolaya Salgado, R., Rojas Pérez, J. C., & González Serna, J. G. (2019). Learning Web Services for E-learning in the Workplace. *IEEE Latin America Transactions*, *17*(11), 1894–1901. https://doi.org/10.1109/TLA.2019.8986429
26. Melgarejo-Torralba, M., Parras-Burgos, D., & Fernández-Pacheco, D. G. (2022). Hand-developed creative prototyping. Methodological proposal and experimentation. *Thinking Skills and Creativity*, *44*, 101025. https://doi.org/10.1016/j.tsc.2022.101025
27. Mercier, M., & Lubart, T. (2023). Video games and creativity: The mediating role of psychological capital. *Journal of Creativity*, *33*(2), 100050. https://doi.org/10.1016/j.yjoc.2023.100050
28. Prasad, R., & Verma, G. (2016). *Software Engineering* (1st ed.). Khanna Book Publishing Co. (P) Ltd.
29. Prendes, M. P., & Gutiérrez, I. (2013). Competencias tecnológicas del profesorado en las universidades españolas. *Revista de Educación*, *361*, 196–222. https://doi.org/10.4438/1988-592X-RE-2011-361-140

30. Rodríguez-García, A.-M., Raso-Sánchez, F., & Ruiz-Palmero, J. (2019). Competencia digital, educación superior y formación del profesorado: un estudio de meta-análisis en la Web of Science. *Píxel-Bit. Revista de Medios y Educación*, *54*, 65–82. https://doi.org/10.12795/pixelbit.2019.i54.04
31. Sandí-Delgado, J. C. (2019). Juegos serios para la indagación de competencias tecnológicas que puedan integrarse en la práctica pedagógica del profesorado. Una propuesta de aplicación en la Sede del Atlántico de la Universidad de Costa Rica (UCR). *TE&ET. Revista Iberoamericana de Tecnología En Educación y Educación En Tecnología*, *23*, 103–105. https://doi.org/10.24215/18509959.23.e13
32. Sandí-Delgado, J. C., & Bazán, P. A. (2017). *Cloud computing: posibilidades para la ejecución de juegos serios educativos as a service (JSEaaS)* [(Tesis de especialización). Universidad Nacional de La Plata (UNLP)]. http://hdl.handle.net/10915/63388
33. Sandí-Delgado, J. C., & Sanz, C. V. (2020). Juegos serios para potenciar la adquisición de competencias digitales en la formación del profesorado. *Revista Educación*, *44*(1), 1–18. https://doi.org/10.15517/revedu.v44i1.37228
34. Sandí-Delgado, J. C., Sanz, C. V., & Lovos, E. N. (2022). Acceptance of Serious Games to Develop Digital Competencies in Higher Education. *The Electronic Journal of E-Learning (EJEL)*, *20*(3), 351–367. https://doi.org/10.34190/ejel.20.3.2181
35. Sanz, C., Gorga, G., González, A., Zangara, A., Iglesias, L., Ibáñez, E., Violini, L., Fachal, A., Archuby, F., Abásolo, M. J., Manresa-Yee, C., Paula, D., & Pesado, P. (2022). Entornos virtuales y herramientas digitales en escenarios educativos híbridos. *XXIV Edición Del Workshop de Investigadores En Ciencias de La Computación*, 744–748. http://sedici.unlp.edu.ar/handle/10915/145252
36. Slivnik, B. (2022). Context-sensitive parsing for programming languages. *Journal of Computer Languages*, *73*, 101172. https://doi.org/10.1016/j.cola.2022.101172
37. Sommerville, I. (2016). *Software Engineering* (M. Borthakur (ed.); 10th ed.). Pearson Education.
38. Spinelli, A. T., & Massa, S. M. (2018). Elicitación de Requerimientos, Centrada en el Usuario, para el Desarrollo de un Serious Game. *XIX Workshop de Investigadores En Ciencias de La Computación*, 1194–1198. http://hdl.handle.net/10915/62882
39. UCR, U. de C. R. (2004). *Consejo Universitario, Acta de la Sesión N.°4932* (pp. 1–30). Consejo Universitario, UCR.
40. Yeh, Y.-F., Hsu, Y.-S., Wu, H.-K., & Chien, S.-P. (2017). Exploring the structure of TPACK with videoembedded and discipline-focused assessments. *Computers & Education*, *104*(January), 49–64. https://doi.org/10.1016/j.compedu.2016.10.006
41. Yulianto, B., Prabowo, H., Kosala, R., & Hapsara, M. (2017). Harmonik = ++(Web IDE). *Procedia Computer Science*, *116*, 222–231. https://doi.org/10.1016/j.procs.2017.10.044
42. Yun, Y.-S., Kim, S., Park, J., Kim, H., Jung, J., & Eun, S. (2020). Development of Neuromorphic Architecture Integrated Development Environment. *International Conference on Green and Human Information Technology*, 47–49. https://doi.org/10.1109/ICGHIT49656.2020.00019

Coordinadores:

- Fernando Bordignon (UNLu)
- Verónica Artola (UNLP)
- Claudia Russo (UNNOBA)

CACiC 2023

CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

XXII Workshop de Tecnología Informática Aplicada en Educación

> Short Papers

WTIAE

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Diseño de sistemas recomendadores en el ámbito de las bibliotecas universitarias

Sergio A. Cabrera, Cristian D. Pacífico y Juan C. L. Teze

Área de Agentes y Sistemas Inteligentes, Facultad de Ciencias de la Administración,
Universidad Nacional de Entre Ríos,
Monseñor Tavella 1424, (E3202KAC) Concordia
{sergio.cabrera, cristian.pacifico, carlos.teze}@uner.edu.ar

Abstract. Esta línea de I+D se ubica dentro del grupo de investigación Agentes y Sistemas Inteligentes de la Facultad de Ciencias de la Administración dependiente de la Universidad Nacional de Entre Ríos (FCAD/UNER).

Persigue como propósito, diseñar un modelo de sistema que utilice técnicas de recomendación y de aprendizaje automático, para realizar sugerencias de fuentes de información, en el ámbito de las bibliotecas universitarias.

En esta etapa de la investigación, se analizarán las técnicas de recomendación, en especial las basadas en contenido, en el ámbito de las bibliotecas universitarias.

Keywords: Sistemas Recomendadores, Bibliotecas Universitarias, Machine Learning.

1. Bibliotecas y Bibliotecas Universitarias

La biblioteca es una organización que tiene como misión fundamental la de otorgar acceso a la información a diferentes usuarios que hacen uso de su acervo documental. Las bibliotecas universitarias se caracterizan por brindar acceso y difusión de fuentes de información, colaborando en los procesos de creación de conocimiento, sirviendo de apoyo a la educación y a la investigación, tanto a la comunidad académica en la que se encuentra inmersa, como así también a la sociedad en general.

La aplicación de la tecnología computacional a las etapas de la actividad científico informativa, ha favorecido la utilización de los Sistemas Integrados de Gestión Bibliotecaria (SIGB).

Se puede definir un SIGB, como un conjunto de programas informáticos diseñados para atender los procesos de las bibliotecas, brindando soluciones a los requerimientos de las mismas, ayudando en los procesos técnicos y en los servicios prestados, favoreciendo la intercomunicación entre bibliotecas. Dentro de estos sistemas se destaca KOHA, por ser el primer SIGB de software libre, permitiendo trabajar con normas de catalogación internacionales, utilizando el formato MARC y el protocolo Z39.50 para la recuperación de información [1].

La Federación Internacional de Asociaciones de Bibliotecarios y Bibliotecas (IFLA)[2], ha enunciado una serie de políticas y recomendaciones con el objetivo de redefinir el trabajo de los bibliotecarios y el rol que las bibliotecas desempeñan en una sociedad que cada día más se amalgama con la Inteligencia Artificial (IA) y en especial con Machine Learning (ML). Las técnicas de IA y ML podrían integrarse con el trabajo

diario del bibliotecario, permitiendo potenciar los servicios y las funciones de las bibliotecas.

2. Sistemas recomendadores

La mayor parte de las actividades que desarrollamos a diario necesitan nutrirse de información para tomar decisiones. El gran volumen de datos e información que se generan diariamente, hacen que esta tarea se torne compleja.

Según Maes, Jannach, Ricci y Melville, como se citó en Briguez [3], los sistemas de recomendación (SSRR) son mecanismos de apoyo que ayudan a los usuarios en su proceso de toma de decisiones al interactuar con volúmenes de información grandes o complejos. La mayoría de los SSRR están destinados a ayudar a los usuarios a hacer frente al problema de la sobrecarga de información, facilitando el acceso a los elementos pertinentes.

Ricci, como se citó en Charnelli [4], establece que la tarea de un Sistema Recomendador (SR) es seleccionar automáticamente los artículos más apropiados para cada usuario de acuerdo a sus intereses y preferencias personales. Generalmente, un SR se centra en un tipo específico de elemento a recomendar denominado "ítem", como por ejemplo un repositorio de materiales educativos, que genera recomendaciones personalizadas para proporcionar sugerencias útiles y efectivas para ese tipo específico de ítem. Se debe tener en cuenta que dependiendo del enfoque de recomendación, según el contexto y la necesidad del usuario, los SSRR generan sugerencias utilizando diversos tipos de conocimiento y datos sobre los usuarios, los ítems disponibles y las transacciones anteriores almacenadas en bases de datos personalizadas.

Según Charnelli, los SSRR se pueden clasificar en cinco categorías dependiendo de la técnica de recomendación: basados en el contenido, filtrado colaborativo, basados en conocimiento, basados en comunidad y por último los híbridos.

Estas técnicas de recomendación pueden mejorarse con las desarrolladas para ML. El ML es un subcampo de las ciencias de la computación y una rama de la IA, cuyo objetivo es desarrollar técnicas que permitan que las computadoras aprendan. El aprendizaje mejora, cuando un sistema logra optimizar su desempeño con la experiencia y el uso de datos, que se convierten en información necesaria para la toma de decisiones.

Los avances en este sentido agregarían nuevas dimensiones y enfoques a los procesos de gestación del conocimiento que se producen en las bibliotecas, en particular, los referidos a la organización, el almacenamiento y la integración de conocimiento. En este sentido se han propuesto soluciones que integran modelos producidos por algoritmos de ML a la construcción de SSRR. De esta forma, se lograron mejorar determinados componentes, haciendo más precisa la recomendación.

Entre algunos desarrollos actuales, es posible citar a Ex Libris Alma [5], como una plataforma unificada de servicios de biblioteca. DARA es el motor inteligente que da soporte a las decisiones de Alma, dotando de IA a las bibliotecas; utiliza algoritmos y análisis inteligentes para adaptar sus recomendaciones a las características de cada una. Además cabe mencionar a Hamlet, que es un sistema que utiliza ML promoviendo interfaces exploratorias experimentales para la colección de Tesis del Instituto

Tecnológico de Massachusetts (MIT). Está en desarrollo y algunas interfaces no se han finalizado todavía. [6][7].

3. Solución propuesta

En primera instancia se pretende caracterizar las técnicas de recomendación, realizando una síntesis de las mismas que asocie a cada una con sus dominios de aplicación más adecuados.

Al particularizar la función de recomendación, se espera obtener un esquema de requisitos y de características deseables de un SSRR de fuentes de información. Luego se realizará una síntesis de las técnicas de ML y su forma de aplicación en el contexto de recomendación de fuentes de información.

3.1 Particularización de la función de recomendación

De la especificación de los SIGB, se identificarán las funcionalidades y los componentes esenciales, resultando de especial interés el estudio de las funciones para el servicio de referencia y el de diseminación selectiva de la información.

En esta etapa de la investigación, se están analizando las técnicas de recomendación en el ámbito de las bibliotecas universitarias, para lo cual se considerarán ciertas particularidades correspondientes a los usuarios y a las fuentes de información.

Se trabajará sobre los usuarios en la categoría de estudiantes. Para confeccionar el perfil de los mismos, serán relevantes los datos provistos por el sistema de gestión académica, SIU-GUARANÍ que se detallan a continuación: edad, carrera, año, materias que cursa, materias aprobadas. Además, será necesario contar con los datos provistos por el historial de préstamos suministrados por SIGB KOHA.

En cuanto a las fuentes de información, se considerarán libros de los cuales serán necesarios los siguientes datos: autor, título, año de edición y cantidad de préstamos realizados durante el último año. Estos datos serán provistos por el SIGB KOHA.

3.1.1 Sistema de recomendación basado en contenido

En estos sistemas, las recomendaciones se basan en el principio de contenido similar. Si un usuario está leyendo un libro, el sistema recomendará otros libros de contenido similar. Se utilizan varios atributos fundamentales para calcular la similitud al verificar contenido semejante.

Esta técnica toma en cuenta algunos datos del historial del usuario para intentar predecir qué busca y cuáles son las sugerencias similares que puede mostrar.

Si bien estos recomendadores presentan algunos inconvenientes, tales como: la sobreespecialización, el análisis de contenido limitado o, en menor medida, el problema de un nuevo usuario (arranque en frío), los mismos deben ser estudiados en el ámbito de las bibliotecas universitarias, dado que se pueden descubrir opciones que se ajustan a las características de los contenidos que se han seleccionado con anterioridad,

permitiendo elegir elementos similares nuevos. Además, los SSRR basados en contenido, son uno de los que presenta mayor presencia en la actualidad.

Para explicar cómo funcionan, se presentará el siguiente caso:

Si un estudiante que se encuentra cursando la asignatura Estructura de Datos, en la carrera de Licenciatura en Sistemas, necesitara solicitar en préstamo un libro de estructura de datos en Java, del autor Mark Allen Weiss; luego de consultar el catálogo de la biblioteca de la Facultad de Ciencias de la Administración, las posibilidades serían:

a. Weiss, Mark Allen. Estructura de datos y algoritmos en Java. Ed. 2000. Cantidad de páginas: 800. Cantidad de préstamos último año: 5.
b. Weiss, Mark Allen. Estructura de datos y algoritmos en Java y C++. Ed. 2013. Cantidad de páginas: 1000. Cantidad de préstamos último año: 100.
c. Weiss, Mark Allen. Estructura de datos en Java. Ed. 2013. Cantidad de páginas: 995. Cantidad de préstamos último año: 102.

Para que el SR pueda sugerir un libro, deberá comprobar la similitud entre los ejemplares. Para determinarla se utilizará el método de distancia euclidiana, en función del año de edición, la cantidad de páginas y la cantidad de préstamos en el último año. Distancia euclidiana entre el libro a y el c:

$$\sqrt{(2000-2013)^2+(800-995)^2+(5-102)^2} = \sqrt{(-13)^2+(-195)^2+(-97)^2} = \sqrt{47603} = 218{,}18$$

Distancia euclidiana entre el libro b y el c:

$$\sqrt{(2013-2013)^2+(1000-995)^2+(100-102)^2} = \sqrt{(0)^2+(-5)^2+(-2)^2} = \sqrt{29} = 5{,}38$$

Se puede observar que la distancia euclidiana es menor entre los ejemplares b y c, que entre los a y c, lo que significa que entre los primeros hay más similitudes que entre los segundos, por lo tanto, el SR, sugeriría los libros b y c.

Para poder explicar el concepto a través de este ejemplo, solo se tomaron algunas características básicas de los libros, pero no existen restricciones para realizar el cálculo de la distancia sobre las características del producto. El principio básico sigue siendo el mismo, si la distancia entre ambos es 0, es probable que tengan un contenido muy similar. Para lograr recomendaciones que se aproximen a las necesidades de los usuarios, lo ideal es combinar diferentes técnicas.

Mathew, Kuriakose y Hegde [8], proponen una estrategia de mejora para los sistemas de recomendación de libros mediante la combinación en cascada de dos métodos: el filtrado basado en contenido y el filtrado colaborativo, logran así que el sistema presente los resultados obtenidos como recomendaciones personalizadas para el usuario. Por su parte, Xin, Haihong, Junde, Meina y Junjie [9], combinan el filtrado colaborativo de vecinos cercanos y el modelo de factor latente. Posteriormente, se exploraron dos modelos para fusionar estas técnicas: la regresión lineal y las redes neuronales. Los resultados indicaron que obtener mejores resultados implicaba combinar distintas técnicas en lugar de implementarlas por separado. Además, destacaron que se lograron predicciones más precisas al emplear redes neuronales en el proceso.

Por otro lado, Tsuji, Kuroo, Sato, Ikeuchi, Ikeuchi, Yoshikane y Itsumura [10], llevaron a cabo un estudio comparativo de tres técnicas de recomendación de libros: el filtrado colaborativo, la minería por reglas de asociación y el algoritmo de Amazon. Su objetivo era determinar cuál de estos métodos era el más efectivo y qué atributos debían ser considerados en el proceso de recomendación. Concluyeron que el algoritmo de Amazon y la minería por reglas de asociación son opciones más efectivas para la recomendación de libros. Se resalta la importancia de considerar los aspectos de privacidad y el costo de los sistemas de recomendación basados en filtrado colaborativo.

De la caracterización de las técnicas de recomendación, se espera lograr una síntesis de las técnicas de recomendación que asocie a cada una con las sugerencias a realizar en el ámbito de las bibliotecas universitarias.

4. Líneas futuras

Luego de realizar un análisis comparativo de las soluciones existentes, realizando pruebas experimentales, se estará en condiciones de diseñar y especificar una arquitectura general para SSRR, proponiendo una guía metodológica para su implementación.

Cabe destacar que la experiencia de un grupo de investigación consolidado en el área de investigación Agentes y Sistemas Inteligentes, habilitará la formación de recursos humanos en la FCAD/UNER, debido a que en paralelo al desarrollo de la investigación, se llevará a cabo la redacción de una tesis de maestría y una tesina de grado.

Referencias

1. KOHA. Consultado el 10 de marzo de 2023. https://koha-community.org/
2. IFLA (2020). IFLA Statement on Libraries and Artificial Intelligence. https://repository.ifla.org/handle/123456789/1646
3. Briguez, C. E. Integración de técnicas cualitativas y cuantitativas en los sistemas de recomendación. [Tesis de Doctor en Ciencias de la Computación, Universidad Nacional del Sur]. Repositorio institucional https://repositoriodigital.uns.edu.ar/bitstream/handle/123456789/4736/BRIGUEZ%2C%20Cristian%20Emanuel%20-%20Tesis%20Doctoral.pdf?sequence =5&isAllowed=y
4. Charnelli, M. E. (2019). Sistemas recomendadores aplicados en educación. [Tesis Especialización en Tecnología Informática Aplicada en Educación, Universidad Nacional de La Plata]. Repositorio institucional http://sedici.unlp.edu.ar/handle/10915/67063
5. ExLibris. Consultado el 10 de febrero de 2023. https://exlibrisgroup.com/
6. American Library Association. (2019). Artificial Intelligence and Machine Learning in Libraries. Library Technology Reports, 55(1). https://doi.org/10.5860/ltr.55n1
7. Yelton, A. (s.f.). Hamlet. Consultado el 18 de febrero de 2023. https://hamlet.andromedayelton.com/
8. Mathew, P., Kuriakose, B. y Hegde, V. (2016). Book recommendation system through content based and collaborative filtering method, International Conference On Data Mining And Advanced Computing (SAPIENCE).

9. Xin, L., Haihong, E., Junde, S, Meina, S. y Junjie, J. (2013). Collaborative book recommendation based on readers borrowing records. International Conference On Advanced Cloud And Big Data.
10. Tsuji, K., Kuroo, E., Sato, S., Ikeuchi, U., Ikeuchi, A., Yoshikane, F. y Itsumura, H. (2012). Use of library loanrecords for book recommendation, IIAI International Conference On Advanced Applied Informatics.

Coordinadores:
- Elsa Estevez (UNS)
- Ariel Pasini (UNLP)

CACiC 2023

CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

VI Track de Gobierno Digital y Ciudades Inteligentes

> Full Papers

TGDCI

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Asistencia posicional para personas ciegas o con baja visión en edificios públicos

Federico Cristina[1] [0000-0003-3838-417X], Sebastián Dapoto[1] [0000-0001-7593-0198], Ramiro Intas[1]

[1] Instituto de Investigación en Informática LIDI, Facultad de Informática,
Universidad Nacional de La Plata, La Plata, Argentina
{fcristina, sdapoto}@lidi.info.unlp.edu.ar, ramiro.intas@hotmail.com

Resumen. El presente trabajo tiene como objetivo contribuir al avance de la gobernanza digital mediante tecnología de asistencia para personas ciegas o con baja visión, brindando una solución innovadora y personalizada para la orientación espacial en entornos no familiares. Mediante la integración de tecnologías móviles y de información geoespacial, se busca mejorar la independencia y la calidad de vida de las personas ciegas o con baja visión u otro tipo de dificultades, como por ejemplo las personas neurodivergentes, facilitando su participación activa en la sociedad y promoviendo la inclusión en entornos académicos y similares. Este trabajo presenta un prototipo de aplicación móvil que contribuye en ese sentido.

Palabras Clave: asistencia posicional en interiores, personas ciegas, baja visión, gobernanza digital.

1. Introducción

En la sociedad actual, la movilidad y la autonomía son elementos fundamentales para el desarrollo pleno de todas las personas. Sin embargo, existen diferentes grupos de personas para las cuales la tarea de desplazarse en entornos desconocidos representa un desafío significativo.

Uno de los grupos más afectados al momento de desplazarse en entornos desconocidos es el de las personas ciegas o con baja visión. Las personas con baja visión pueden disponer de cierta visión residual, pero no suficiente para ver objetos a distancia o leer carteles. Asimismo, su visión puede ser afectada por el exceso o la falta de iluminación, o pueden sufrir otro tipo de patologías como visión borrosa o nistagmo, afectando no sólo su visión, sino también su equilibrio y coordinación.

Por otro lado, existen otros grupos de personas que pueden necesitar ayuda al momento de orientarse en entornos desconocidos, como por ejemplo las personas neurodivergentes. Las personas neurodivergentes [1] presentan una actividad cerebral diferente, cuyas habilidades y formas de procesar la información no se ajustan a lo que se considera común o estándar. Algunos ejemplos de neurodivergencia incluyen el autismo, el TDAH, la dislexia, el Síndrome de Tourette, entre otros. Las personas que tienen alguna de estas condiciones, pueden experimentar ansiedad social, es decir, que el solo hecho de entablar una conversación con un desconocido, por ejemplo, para pedir información de ubicación, puede generarle un nivel alto de estrés.

La falta de acceso a información espacial precisa y la dificultad para orientarse en lugares nuevos limitan la independencia de las personas ciegas o con baja visión y personas neurodivergentes, pudiendo generar situaciones de ansiedad y frustración.

Uno de los contextos en los que esta problemática se hace más evidente es en las instalaciones de instituciones como facultades y campus universitarios, entre otros. Estos espacios suelen ser extensos y complejos, con múltiples edificios, aulas y áreas comunes. Para las personas de los grupos mencionados anteriormente o visitantes que no están familiarizados con la disposición del lugar, movilizarse y encontrar ubicaciones específicas puede resultar un reto abrumador.

El desafío de brindar asistencia posicional a personas ciegas o con baja visión y con otro tipo de dificultades en edificios públicos va más allá de la mera implementación tecnológica. Es necesario considerar el concepto de gobernanza digital, que implica la adopción de políticas, normativas y prácticas que garanticen la accesibilidad y la inclusión de todas las personas en el entorno construido [2] [3]. La gobernanza digital en edificios públicos implica la integración de tecnologías accesibles y la colaboración entre diferentes actores, como los desarrolladores de aplicaciones, las instituciones educativas y las autoridades gubernamentales, para asegurar que las soluciones tecnológicas sean implementadas de manera efectiva y sostenible.

Los avances tecnológicos recientes ofrecen nuevas oportunidades para abordar este problema [4] y mejorar la independencia y la calidad de vida de las personas ciegas o con baja visión. En particular, el desarrollo de aplicaciones móviles y la integración de tecnologías como la geolocalización y la síntesis del habla brindan la posibilidad de crear soluciones innovadoras que asistan a las personas en la ubicación y desplazamiento en entornos desconocidos [5] [6].

El objetivo de este trabajo es presentar un enfoque integral para abordar la problemática de la orientación espacial de las personas ciegas o con baja visión en edificios públicos. Se explorarán las diversas tecnologías disponibles y se propondrá el desarrollo de un prototipo móvil personalizado que ofrezca asistencia auditiva a las personas, proporcionándoles información sobre su ubicación y guiándolos de manera efectiva hacia destinos específicos. El objetivo final del proyecto es convertir el prototipo en una aplicación móvil de asistencia para personas ciegas en el contexto específico de las instalaciones de la Facultad de Informática de la Universidad Nacional de La Plata (UNLP).

Además de presentar el marco teórico y las soluciones propuestas, se abordarán aspectos prácticos relacionados con la usabilidad, la accesibilidad y la aceptación de la tecnología por parte de los usuarios. Se discutirán también los desafíos y las limitaciones que pueden surgir al implementar estas soluciones en entornos reales, así como las posibles vías para futuras investigaciones y mejoras en esta área.

El resto del trabajo se organiza de la siguiente manera: en la sección 2 se detallan los desafíos que enfrentan las personas con problemas de visión, así como la importancia de una solución de asistencia posicional. La sección 3 describe la metodología utilizada para desarrollar la aplicación, incluyendo las generalidades del algoritmo de entrenamiento y estimación posicional utilizados. En la sección 4 se detalla la solución desarrollada, así como su arquitectura y funcionalidades relacionadas. La sección 5 presenta los escenarios de pruebas, los resultados obtenidos y el análisis de la precisión de la asistencia posicional brindada por la aplicación. Finalmente, en la sección 6 se resumen las principales contribuciones del trabajo, las limitaciones encontradas y se presentan posibles líneas de mejora para futuras investigaciones.

2. Motivación

La asistencia posicional en interiores para personas ciegas o con baja visión es un desafío importante en la mejora de su autonomía y seguridad en entornos desconocidos. La movilidad en espacios cerrados, como edificios públicos, centros comerciales o instituciones educativas, puede resultar especialmente complicada para las personas ciegas o con baja visión. La falta de referencias y la dificultad para orientarse en estos entornos puede generar frustración y limitar la participación plena en la sociedad. En este contexto, la tecnología ofrece oportunidades para desarrollar soluciones innovadoras que ayuden a las personas a desplazarse y ubicarse de manera efectiva en espacios interiores.

La triangulación WiFi se presenta como una tecnología ideal para abordar el desafío de la asistencia posicional en espacios interiores [7][8]. Esta técnica se basa en la medición de la intensidad de señales WiFi provenientes de puntos de acceso distribuidos en el entorno. Al analizar las diferencias de intensidad de señal entre los puntos de acceso detectados por un dispositivo móvil, es posible determinar la ubicación relativa del usuario en el espacio. La ventaja de utilizar WiFi radica en su amplia disponibilidad en entornos interiores, lo que permite una cobertura extensa y una precisión razonable para la asistencia posicional.

Es importante destacar que la precisión de la triangulación WiFi puede verse afectada por diversos factores, como la interferencia de otras señales, los obstáculos en el entorno, la variabilidad de la intensidad de la señal y la configuración de los puntos de acceso. Sin embargo, con un mapeo adecuado y algoritmos robustos, se puede lograr una asistencia posicional útil para personas ciegas o con baja visión en entornos interiores.

Otro punto fundamental del proyecto es la tecnología de síntesis del habla, la cual desempeña un papel crucial en la comunicación efectiva con las personas ciegas o con baja visión [9]. Mediante el uso de algoritmos de síntesis del habla se transforma la información de ubicación obtenida mediante la triangulación WiFi en indicaciones auditivas claras y comprensibles. Estas indicaciones se transmiten en tiempo real al usuario a través de auriculares o altavoces integrados en el dispositivo móvil, brindándole orientación precisa sobre su posición actual y las direcciones a seguir. La síntesis del habla no sólo proporciona información práctica para el desplazamiento, sino que también permite una interacción natural y fluida entre la persona ciega o con baja visión y el prototipo móvil, mejorando la experiencia del usuario y fomentando su independencia en entornos interiores desconocidos.

El presente trabajo tiene como objetivo desarrollar un prototipo móvil que utilice la triangulación WiFi para brindar asistencia posicional en espacios interiores a personas ciegas o con baja visión. El prototipo permitirá a los usuarios obtener información sobre su ubicación actual, así como recibir indicaciones auditivas para desplazarse de manera segura y eficiente en entornos desconocidos. Este tipo de asistencia será de utilidad no solo para las personas ciegas o con baja visión, sino también para otros grupos de personas como, por ejemplo, las personas neurodivergentes.

Para lograr el objetivo propuesto, se realiza un análisis de los algoritmos de triangulación WiFi para implementar un sistema robusto que garantice una precisión adecuada.

3. Metodología

Si bien existen varias opciones de posicionamiento en interiores mediante intensidades de señales WiFi, una de las más utilizada por su simplicidad, velocidad de respuesta y bajo costo computacional es la de fingerprinting [10].

El proceso de implementación y uso de esta técnica involucra básicamente las siguientes etapas y pasos:

1. Entrenamiento: lograr contar con un conjunto de valores de referencia de intensidades de señal WiFi a lo largo de la zona de interés. Esto implica:
    a. Recopilar datos de intensidad de señal WiFi en diferentes ubicaciones en el área de interés.
    b. Almacenar los datos de intensidad de señal y la ubicación correspondiente en una base de datos.
2. Estimación: determinar la posición del dispositivo móvil basándose en los datos obtenidos en el entrenamiento previo. Esto implica:
    a. Escanear los puntos de acceso WiFi cercanos y recopilar sus intensidades de señal.
    b. Consultar la base de datos para encontrar los puntos de referencia más cercanos en función de las intensidades de señal escaneadas.
    c. Utilizar la ubicación de los puntos de referencia más cercanos para estimar la ubicación del dispositivo.

En la etapa de entrenamiento, se lee la intensidad de las señales de cada BSSIDs (direcciones MAC) de los puntos de acceso WiFi correspondientes a una ubicación en particular, almacenando dicha información para luego ser utilizada en la etapa de estimación.

En la etapa de estimación, el algoritmo del punto 2.c básicamente implica realizar la consulta a la base de datos para obtener las intensidades de cada BSSID almacenados e iterar a través de éstos calculando la diferencia entre las intensidades de señal escaneadas y las intensidades de señal almacenadas en cada punto de referencia, realizando esta misma actividad para cada ubicación almacenada. Finalmente se determina la posición tomando la ubicación que presente la menor diferencia entre todos los puntos de referencia.

4. Diseño del prototipo móvil

A fin de poder llevar adelante pruebas de campo para la solución propuesta, se desarrolló un prototipo para dispositivos móviles con sistema operativo Android [11].

Android se ha consolidado como el sistema operativo más popular y ampliamente utilizado en dispositivos móviles en todo el mundo, con una amplia base de usuarios significativamente mayor que en otras plataformas, como iOS para iPhone. La cuota de mercado mundial de Android es mayor al 70% [12], y si solo se tiene en cuenta a Argentina, este valor se eleva al 90% [13], por lo cual se convierte en una elección sólida para el desarrollo del prototipo, ya que brinda la oportunidad de llegar a una audiencia más amplia.

Además de su popularidad, Android se destaca por su accesibilidad y flexibilidad. El sistema operativo ofrece una amplia gama de herramientas y recursos de accesibilidad integrados, lo que permite a los desarrolladores crear aplicaciones inclusivas y adaptadas a las necesidades de las personas ciegas o con baja visión. Android también ofrece una mayor libertad en términos de personalización y configuración, lo que permite adaptar la experiencia de usuario y las funcionalidades del prototipo móvil de manera más flexible y precisa.

El prototipo consta básicamente de dos modos de uso:

- Modo Administrador: este modo permite realizar el entrenamiento preliminar necesario para contar con la base de conocimiento referencial conteniendo los niveles de intensidad de todos los puntos de acceso WiFi. El entrenamiento simplemente implica especificar la ubicación actual y ejecutar el escaneo de señales WiFi para luego almacenar los valores resultantes. Asimismo, es posible realizar automáticamente el escaneo cierto número de veces a fin de disminuir el potencial ruido y obtener mediciones más precisas. La variabilidad en la intensidad de las señales WiFi puede estar influenciada por diversos factores, como la interferencia electromagnética y la presencia de obstáculos físicos. Al realizar múltiples mediciones en diferentes momentos y ubicaciones, podemos capturar una gama más amplia de datos y obtener una imagen más completa de las características de las señales en cada ubicación. Este enfoque permite filtrar el ruido y obtener mediciones más estables y confiables. La figura 1.a muestra el modo administrador.
- Modo Asistencia: este modo es el que utilizan los usuarios de la aplicación y permite consultar su posición actual. Al presionar el botón de consulta, se dispara un escaneo de intensidad de señales WiFi y se determina la posición en función de los datos generados en el entrenamiento, tal como se detalló en el apartado anterior. La figura 1.b muestra el modo asistencia.



a) b)

Fig. 1. a) Modo administrador. b) Modo asistencia.

La definición estructural que se almacena es básicamente la que se describe a continuación:

- Ubicaciones: esta propiedad es una lista de:
  - Nombre de ubicación ○ Intensidades, esta propiedad es una lista de:
    - BSSID
    - Intensidad de señal

En el modo administrador la información del entrenamiento se almacena en formato JSON. El prototipo desarrollado permite exportar el entrenamiento realizado, así como importarlo en otro dispositivo en caso de que sea de utilidad reutilizar un entrenamiento previo. A modo de ejemplo, la figura 2 muestra de manera reducida el contenido almacenado como resultado de la etapa de entrenamiento.

```
{
  "locations" : [
    {
      "name" : "oficina",
      "scanDetails" : [
        {
          "bssid" : "d8:07:b6:ae:a4:a6",
          "ss" : 99
        },
        {
          "bssid" : "b0:ec:dd:ec:89:c4",
          "ss" : 17
        }
      ]
    },
    {
      "name" : "estudio",
      "scanDetails" : [
        {
          "bssid" : "d8:07:b6:ae:a4:a6",
          "ss" : 76
        },
        {
          "bssid" : "6a:c6:3b:83:9a:5c",
          "ss" : 37
        }
      ]
    }
  ]
}
```


Fig. 2. Estructura resultante de la etapa de entrenamiento.

5. Experimentación y resultados obtenidos

El uso de señales WiFi puede presentar ciertas limitaciones que deben ser consideradas. En primer lugar, la precisión de la triangulación WiFi puede verse afectada por factores externos, como la interferencia de otras redes WiFi cercanas o cambios en el entorno físico. Esto podría generar mediciones inconsistentes y, en consecuencia, afectar la precisión de la determinación de la ubicación. Además, la solución depende de la disponibilidad y calidad de las redes WiFi en el entorno. En áreas con una cobertura WiFi deficiente o con redes poco estables, la precisión y confiabilidad de la asistencia posicional pueden verse comprometidas. Esto puede limitar la aplicación en entornos donde la infraestructura de redes WiFi no esté completamente desarrollada.

A fin de validar la correcta usabilidad del prototipo móvil desarrollado, se diseñaron y ejecutaron una serie de pruebas preliminares en ambientes controlados. Para la realización del set de pruebas se utilizó un edificio residencial de la ciudad de La Plata, ubicado en el casco urbano. Este edificio cuenta con cuatro plantas y 12 departamentos

de aproximadamente 35 metros cuadrados en cada una de ellas. El edificio ocupa una superficie total de 18,44m x 58m.

En primera instancia se realizó la etapa de entrenamiento de la aplicación, por lo que se procedió a escanear las señales WiFi en las distintas áreas del espacio cerrado para guardarlas en la base de datos y utilizarlas como referencia. Para esto, se tomaron 3 puntos de referencia en cada planta del edificio. Las ubicaciones escaneadas en la etapa de entrenamiento fueron las siguientes:

- Planta Baja: izquierda, medio, derecha, estacionamientos 1 y 2.
- Piso 1: izquierda, medio y derecha. • Piso 2: izquierda, medio y derecha.
- Piso 3: izquierda, medio y derecha.

En las figuras 3 y 4 es posible observar un esquema de las diferentes plantas del edificio, en donde aparecen marcadas las ubicaciones escaneadas como puntos rojos.

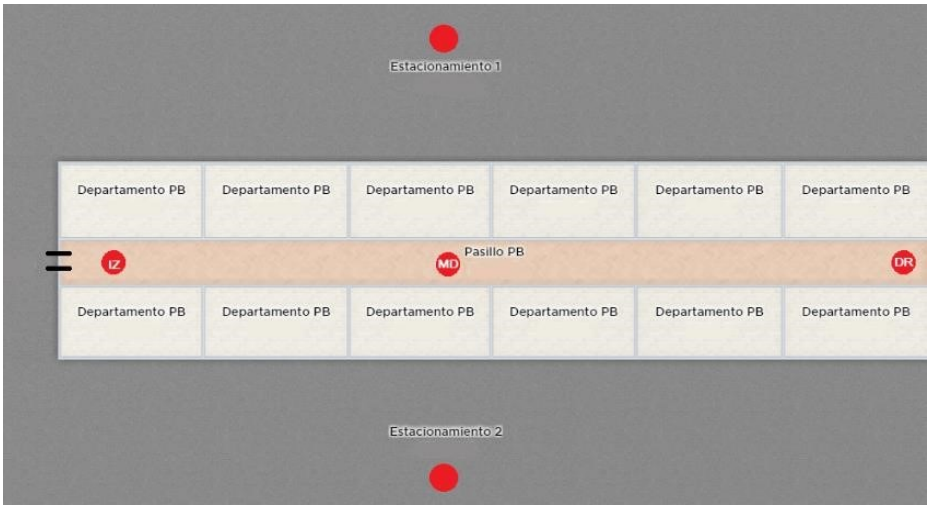


Fig. 3. Ubicaciones escaneadas en la planta baja.

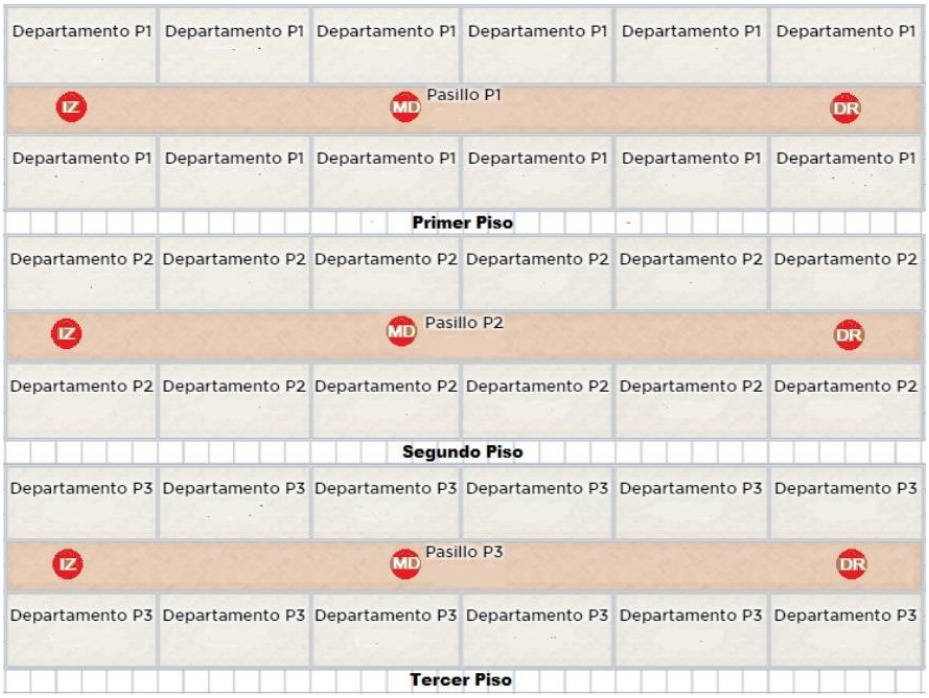


Fig. 4. Ubicaciones escaneadas en las plantas del primer, segundo y tercer piso.

Se realizó una serie de recorridos al edificio, transitando por las diferentes áreas y verificando la posición informada por la aplicación. En la tabla 1 se muestra el porcentaje de efectividad en las distintas áreas del edificio. El promedio de aciertos del total de las pruebas es de 78,57%, y la mediana es de 91,67%. Se debe tener en cuenta que dentro de estos resultados se encuentran dos secciones al aire libre que no cuentan con una red WiFi cercana, y generan más errores que aciertos. Si no se tienen en cuenta esos dos sectores al aire libre, el promedio de aciertos asciende a 86,81%, y la mediana a 100%.

Adicionalmente a las pruebas de ubicación, se verificaron aspectos no funcionales pero necesarios para este tipo de aplicaciones, tales como la facilidad de uso y la adecuada asistencia auditiva correspondiente. Cabe mencionar además que, durante todo el proceso de desarrollo, se ha mantenido una estrecha colaboración con expertos en el campo de la disminución visual, quienes han brindado valiosos aportes para determinar las opciones de implementación más adecuadas.

El próximo paso luego de haber realizado estas pruebas preliminares será el de efectivizar el entrenamiento en el edificio de la Facultad de Informática de la UNLP, con la finalidad de realizar pruebas exhaustivas en el ambiente real y liberar la solución para cualquier persona que así lo requiera. Además, se planea llevar a cabo una evaluación exhaustiva mediante pruebas con usuarios reales para validar su efectividad y usabilidad.

Tabla 1. Resultados obtenidos en las diferentes áreas testeadas.

Piso y sección	Porcentaje de efectividad
PB/Izquierda	100,00%
PB/medio	75,00%
PB/derecha	50,00%
P1/Izquierda	100,00%
P1/medio	83,33%
P1/derecha	100,00%
P2/Izquierda	100,00%
P2/medio	100,00%
P2/derecha	66,67%
P3/Izquierda	100,00%
P3/medio	66,67%
P4/derecha	100,00%
Estacionamiento 1	33,33%
Estacionamiento 2	25,00%

6. Conclusiones y trabajos futuros

En el presente artículo se ha abordado la problemática de las personas ciegas o con baja visión al ubicarse en entornos no familiares, y se ha desarrollado un prototipo móvil que utiliza tecnologías de asistencia posicional para brindarles apoyo en su desplazamiento.

La gobernanza digital en edificios públicos es fundamental para garantizar la accesibilidad y la inclusión de las personas ciegas o con baja visión. La colaboración entre desarrolladores de aplicaciones, instituciones educativas y autoridades gubernamentales es esencial para implementar soluciones tecnológicas accesibles de manera efectiva y sostenible.

La tecnología puede desempeñar un papel crucial en la mejora de la autonomía y la seguridad de las personas ciegas o con baja visión en entornos interiores desconocidos. La triangulación WiFi se ha mostrado como una tecnología ideal para determinar la ubicación en entornos interiores. La combinación de algoritmos de triangulación y técnicas de procesamiento de señales ha permitido obtener resultados precisos y confiables en la determinación de la posición.

La investigación realizada y el desarrollo del prototipo móvil han mostrado el potencial de las tecnologías de asistencia posicional para mejorar la vida de las personas ciegas o con baja visión y otros grupos, como las personas neurodivergentes, al desplazarse en entornos interiores desconocidos. Además, se ha resaltado la importancia de la gobernanza digital en la creación de entornos inclusivos y accesibles. Se espera que este trabajo inspire a futuras investigaciones y promueva la implementación de soluciones tecnológicas que brinden mayor autonomía y calidad de vida a las personas ciegas o con baja visión.

Se espera realizar el entrenamiento de la aplicación móvil en el edificio de la Facultad de Informática de la UNLP, efectuando pruebas exhaustivas con usuarios reales para validar su efectividad y usabilidad.

A futuro se considera la implementación de algoritmos de entrenamiento y estimación más sofisticados para el procesamiento de datos de señales WiFi, lo cual podría permitir una mejor interpretación de los patrones y una mayor capacidad de adaptación a cambios en el entorno.

Adicionalmente se contempla la posibilidad de implementar asistencia mediante otras tecnologías, como por ejemplo el uso de Balizas Bluetooth o el uso de imágenes y/o video para mejorar la precisión y la robustez de la asistencia posicional. La integración de estas tecnologías podría proporcionar una mayor redundancia y permitir una mejor adaptación a diferentes entornos.

Referencias

1. Amador Fierros, Genoveva et al. Neurodiversity in Higher Education: the students experience. Rev. educ. sup [online]. 2021, vol. 50, n. 200, pp. 129-151. Epub 22-Mar-2022. ISSN 0185-2760. https://doi. org/10.36857/resu. 2021.200. 1893.
2. Revista Bit & Byte. Año 3, N° 6. "La gobernanza digital, un concepto más amplio que el e-gobierno". Revista Institucional de la Facultad de Informática, Universidad Nacional de La Plata. ISSN: 2468-9564. 2017.
3. R. Gacitúa, H. Astudillo, B. Hitpass, M. Osorio-Sanabria and C. Taramasco, "Recent Models for Collaborative E-Government Processes: A Survey," in IEEE Access, vol. 9, pp. 19602-19618, 2021, doi: 10.1109/ACCESS.2021.3050151.
4. A. Hameed and H. A. Ahmed, "Survey on indoor positioning applications based on different technologies," 2018 12th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics (MACS), Karachi, Pakistan, 2018, pp. 1-5, doi: 10.1109/MACS.2018.8628462.
5. J. P. Gomes, J. P. Sousa, C. R. Cunha and E. P. Morais, "An indoor navigation architecture using variable data sources for blind and visually impaired persons," 2018 13th Iberian Conference on Information Systems and Technologies (CISTI), Caceres, Spain, 2018, pp. 1-5, doi: 10.23919/CISTI.2018.8399347.
6. L. Árvai, "Mobile phone based indoor navigation system for blind and visually impaired people: VUK — Visionless supporting frameworK," 2018 19th International Carpathian Control Conference (ICCC), Szilvasvarad, Hungary, 2018, pp. 383-388, doi: 10.1109/CarpathianCC.2018.8399660.
7. S. Qiao, C. Cao, H. Zhou and W. Gong, "The trip to WiFi indoor localization across a decade — A systematic review," 2023 26th International Conference on Computer Supported Cooperative Work in Design (CSCWD), Rio de Janeiro, Brazil, 2023, pp. 642647, doi: 10.1109/CSCWD57460.2023.10152700.
8. R. Hegde, S. K. Hegde, K. Prasad, V. Srinivas, T. De and V. Dankan Gowda, "Wi-Fi Router Signal Coverage Position Prediction System using Machine Learning Algorithms," 2023 International Conference on Sustainable Computing and Smart Systems (ICSCSS), Coimbatore, India, 2023, pp. 253-258, doi: 10.1109/ICSCSS57650.2023.10169501.
9. F. Brandan, D. Pandolfi, A. Villagra, M. Errecalde. "Sistemas conversacionales aplicados a la gobernanza". XXIV Workshop de Investigadores en Ciencias de la Computación (WICC 2022, Mendoza). ISBN: 978-987-48222-3-9. Páginas: 87-91. Año 2022.
10. Y. Dong, T. Arslan, Y. Yang and Y. Ma, "A WiFi Fingerprint Augmentation Method for 3D Crowdsourced Indoor Positioning Systems," 2022 IEEE 12th International Conference on

Indoor Positioning and Indoor Navigation (IPIN), Beijing, China, 2022, pp. 1-8, doi: 10.1109/IPIN54987.2022.9918117.
11. Github Proyecto UPA. https://github.com/fedecris/unsighted-position-assist.
12. Mobile Operating System Market Share Worldwide. https://gs.statcounter.com/os-marketshare/mobile/worldwide.
13. Mobile Operating System Market Share Argentina. https://gs.statcounter.com/os-marketshare/mobile/argentina.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

La gobernanza en los Ecosistemas Digitales de Integrabilidad a la luz de la innovación tecnológica

Giorgetti Gustavo[1], Luz Clara Horacio[1] , Bazán Patricia[2]

[1] Facultad de Ingeniería - Universidad FASTA

[2] LINTI, Facultad de Informática, UNLP

ggiorgetti@ufasta.edu.ar; hluzclara@ufasta.edu.ar; pbaz@info.unlp.edu.ar

Resumen. La transformación digital tiene lugar gracias a la participación de factores facilitadores y habilitantes tecnológicos que es posible combinar sinérgicamente para arribar a modelos, usos y resultados innovadores. La aparición de los EDI (Ecosistema Digital de Integrabilidad) ha posibilitado nuevas formas de vinculación organizacional y a partir de ello, respuestas originales para las necesidades humanas. Esta realidad determina cambios profundos en la manera de entender los intercambios y las relaciones entre instituciones en general, por lo cual resulta vital asegurar su gobernanza. Además de las relaciones interorganizacionales, quedan bajo la gobernanza EDI una diversidad de materias que hacen a la información y su manipulación por medios tecnológicos. Dichos aspectos, como, por ejemplo, los relativos a datos, procesos y servicios, podrían considerarse en sí mismos objetos de gobernanzas particulares, que se imbrican y aparecen encarnados en tecnologías tradicionales y no tradicionales que deben encauzarse. Este trabajo analiza el impacto potencial de la utilización de algunas nuevas tecnologías en el contexto EDI y propone pautas que sirvan a un posterior desarrollo de marcos de trabajo y modelos de gobernanza que las contengan.

Palabras Clave: Gobernanza EDI, Gobernanza de datos, Gobernanza de procesos, Gobernanza TIC, Gobernanza de la información, Tecnologías transversales, Ecosistema Digital de Integrabilidad, IRAM 17610-1.

1. Introducción

El empleo de medios tecnológicos y su constante recombinación permite dar mejores y más eficientes respuestas a las necesidades humanas. Con tal perspectiva, organizaciones públicas y privadas encuentran en la digitalización una vía apta para facilitar el logro de sus metas, asimilando progresivamente nuevas formas de trabajo conducentes a maximizar el aprovechamiento de dichas herramientas tecnológicas.

El avance de estas posibilidades trasciende hoy en día los límites de una organización, para conformar nuevas y más extensas cadenas de valor fundadas en el intercambio de información entre diversos productores y consumidores de la misma.

La aparición de los EDI discurre en tal sentido, pero apuntando no simplemente a un mero consumo de elementos existentes por nuevos actores, sino, y particularmente, a la creación, a partir de aquellas piezas, de nuevos servicios que generan valor antes no disponible. Entendidos de este modo, los EDI aportan una concepción integral para la construcción de soluciones digitales interorganizacionales.

A nivel organizacional, la interoperabilidad puede ser adecuada y estructurada a través de marcos apropiados, pero resulta insuficiente a nivel interorganizacional.

En línea con lo enunciado, una técnica innovadora es lograr reutilizar datos, complementar procesos organizacionales y mezclar ambos mundos (datos y procesos) para producir nuevos resultados. Esta capacidad de poder integrar/unificar servicios sobre la interoperabilidad da origen al concepto de integrabilidad.

Un EDI (Ecosistema Digital de Integrabilidad), descripto y caracterizado en [Bazan et al, 2022], es una comunidad de organizaciones miembros que: 1- respeta mínimas reglas de convivencia digital, 2- aplica estándares y componentes de software para poder utilizar y reutilizar los servicios comunes del ecosistema.

Un EDI debe habilitar la innovación mucho más allá de la interoperabilidad y toda norma que la soporte debe facilitar y potenciar dicha innovación.

La norma argentina IRAM 17610 Ecosistema digital de integrabilidad. Parte 1 – Requisitos [IRAM 17610-1, 2023], constituye un documento técnico establecido por consenso y aprobado por un organismo reconocido que proporciona, para usos comunes y repetidos, reglas, directrices o características para las actividades o sus resultados, a fin de garantizar un nivel óptimo de orden en un contexto dado. Establece los requisitos técnicos, organizacionales y de gestión, así como recomendaciones para un EDI y define la *gobernanza* y el *urbanismo digital*.

Dentro del enfoque de la norma, la gobernanza involucra a quienes toman las decisiones, cuáles son los tipos de decisiones que se toman y además atraviesa las cuatro dimensiones sobre las cuales se definen requisitos o recomendaciones para un EDI, en dicha norma. (legal, organizacional, semántica y técnica).

En este sentido, los antecedentes aportados por la norma IRAM 17610-1, constituyen una base para el análisis que se propone en este trabajo acerca del impacto que las nuevas tecnologías tienen en los EDI.

El trabajo se organiza de la siguiente manera: en la sección 2 se abordan los propósitos y desafíos de la gobernanza de un EDI y sobre ello, en la sección 3 se analizan los efectos de la evolución de las tecnologías en el contexto de dicha gobernanza. En la sección 4, presenta la gobernanza de un EDI en torno a datos, procesos y servicios, para luego discutir acerca de la gobernanza de un EDI y la transversalidad tecnológica. Por último, la sección 6 presenta las conclusiones.

2. Propósito y desafíos de la gobernanza de un EDI

La gobernanza de un Ecosistema Digital de Integrabilidad (EDI), tiene por propósito: lograr un despliegue y desarrollo del EDI en forma continua y sustentable, donde: a- la experiencia del ser humano en sus distintos roles (ciudadano, paciente, emprendedor, etc.) sea en todo momento la mejor posible; b- todas las organizaciones miembros puedan desarrollar sus propios propósitos más eficazmente dentro del EDI que fuera de él; c- se logre minimizar la duplicación de desarrollos y esfuerzos, buscando optimizar y estandarizar los aspectos comunes y permitir la diversidad para los temas particulares; d- se logre minimizar el tiempo y costo en el desarrollo del EDI; e- se logre el mejor aprovechamiento posible tanto en la reutilización inteligente de los desarrollos existentes como de las nuevas tecnologías emergentes.

A partir del propósito enunciado, se pueden identificar algunos desafíos claves y recurrentes de la gobernanza de un EDI, especialmente: 1- La diversidad de intereses: se debe poder contemplar la diversidad de intereses y miradas de las múltiples organizaciones involucradas sin caer en decisiones impuestas por la mayoría que dejan aspectos sin atender o considerar, lo cual implicaría potenciales riesgos y/o futuros problemas; 2- El todo y las partes: mantener coherente el desarrollo del EDI como un todo, donde actores, tecnologías y los propios conocimientos humanos se abren paso en partes cada vez más especializadas, que llegan incluso a redescubrir principios y modelos que ya fueron definidos en otras y que deberían ser de uso común por todos; 3- Lo urgente versus lo importante: se deben tomar decisiones estratégicas, tácticas y operativas que estén alineadas con el desarrollo sustentable del EDI. Seguramente se deberán atender urgencias a través de soluciones coyunturales, pero es la gobernanza la que con sus decisiones estratégicas deberá resolver este desafío.

Estos desafíos de la gobernanza de un EDI surgen de la propia naturaleza multi-organizacional, con alta diversidad tecnológica, donde todos sus miembros deben encontrar en conjunto, colaborativamente y sin la supremacía de ninguno, el mejor proceso de transformación digital para el despliegue sustentable del mismo.

3. La gobernanza EDI y las TIC

Toda organización, para poder desarrollar y lograr sus propósitos necesita relacionarse con otras organizaciones. Estas relaciones son necesidades básicas para su funcionamiento y son independientes de las soluciones tecnológicas que finalmente se seleccionen y utilicen para satisfacerlas.

Si se clasifican estas necesidades en función de su naturaleza, se pueden encontrar tres que se repiten en todos los casos, estas son: 1- *compartir datos*, que se vincula con la necesidad de las organizaciones de contar con datos de calidad, 2- *co-crear procesos*, que habilita a las organizaciones a coordinarse entre sí, para aportar a una cadena de valor y 3- *liberar servicios*, que permite a las organizaciones combinar o integrar servicios puestos a disposición por distintos actores.

Las tecnologías de la información a través de las cuales se concretan aspectos organizacionales y técnicos, se renuevan y suceden de continuo, a la par que amplían su alcance.

Un primer efecto de esta realidad es una creciente e ineludible dependencia de los medios electrónicos, de lo cual inmediatamente se sigue que los EDI deben interpretarse como sistemas de misión crítica, por lo cual deben gobernarse gestionando sus riesgos en consonancia, para asegurar un funcionamiento estable e ininterrumpido. Con todo, se trata de una situación que, lejos de resultar exclusiva, penetra cada vez más en todo el mundo informatizado.

Un segundo efecto se observa en la tendencia a la incorporación de nuevas tecnologías antes de que éstas alcancen madurez, lo que resulta en una visión superficial de la totalidad de sus efectos potenciales, tanto favorables como desfavorables, más allá de una utilización subóptima de la inversión previa en dicho rubro. Cabe señalar que la filosofía de reutilización que es central a los EDI contribuye a contrarrestar esta ineficiencia, aunque subsiste la necesidad de una consideración cuidadosa a nivel de

gobernanza que anticipe las repercusiones intra e interorganizacionales de la introducción de tecnologías no convencionales.

Un tercer efecto está dado por el rezago legal y regulatorio sobre nuevas tecnologías de la información, lo que mantiene abierto un horizonte temporal no determinado para usos eventualmente abusivos, sean éstos intencionales o inadvertidos.

Los efectos anteriormente mencionados subrayan la necesidad de un gobierno de la tecnología, pero especialmente de las más radicales y poco conocidas, a través de mecanismos flexibles y ágiles, que acompañen la innovación a la vez que aseguren paridad y participación en las decisiones para todos los participantes de un EDI. Estas reglas mínimas de funcionamiento contribuyen a la generación de confianza entre miembros más allá del peso específico o grado de sofisticación que ostenten.

Aquí la norma IRAM 17610-1 asegura que el despliegue y operación sustentable de un EDI sea posible. Un EDI donde convivirán digitalmente distintas organizaciones que, si bien comparten las mismas necesidades arriba mencionadas, podrán tener distintos intereses, algunas veces contrapuestos, o prioridades diferentes. La situación resulta similar al problema de la urbanización en el mundo físico, por lo cual se habla de urbanismo digital.

Se hace evidente la importancia de contar con una gobernanza sobre ese espacio digital común (inter-organizacional). Una gobernanza que tome decisiones que todos los miembros del EDI deben acatar en el espacio común, respecto a las tres necesidades definidas: compartir datos, cocrear procesos y liberar servicios.

4. Gobernanza EDI: datos, procesos y servicios

Considerada en su conjunto, la gobernanza EDI puede entenderse en su máxima expresión, como una gobernanza de las relaciones interorganizacionales, al estilo de la gobernanza de red, bajo la cual discurre una gobernanza de la información y de la tecnología que la soporta. Su núcleo duro se centra así en datos, procesos y servicios compartidos y es atravesada por una serie de transversalidades, en las cuales destacan los aspectos regulatorios y legales, y el gobierno de la seguridad, la resiliencia y el riesgo.

En relación a las tecnologías de más reciente difusión, la creciente convergencia de conceptos y herramientas informáticas determina que su gobernanza, más allá de cualquier consideración atinente a tecnologías particulares, deba entenderse en términos de su articulación conjunta. Es así como puede visualizarse la imagen completa de sus posibilidades y sus riesgos, y obtener, subsiguientemente, reglas para su operación. Corresponde a la gobernanza EDI garantizar un uso responsable de las nuevas tecnologías, en lo cual no pueden dejar de contemplarse derechos humanos básicos, como el derecho a la intimidad, el trato no discriminatorio o la libre expresión.

Más allá de la falta de acuerdo y la proliferación de principios a escala global en torno a las nuevas tecnologías [Kavanagh, 2019], la realidad es que la implementación efectiva de un EDI no puede más que partir de sus propios valores fundacionales y principios guía. Son puntualmente estos últimos los que, enfocándose en aspectos éticos, de transparencia, o de responsabilidad, entre otros, ofrecen las directrices de las cuales derivan políticas y objetivos estratégicos para transmitir al nivel de gestión y

posteriormente poder evaluar su logro. Debe tomarse nota de que la formulación de principios desde un posicionamiento meramente teórico puede ser ocasión de desacuerdo, y aún no siendo así, aquéllos requieren de todos modos de una operacionalización que habilite su cumplimiento efectivo en la práctica. Por tal razón, este trabajo presenta, a través de consideraciones puntuales, elementos para la extracción de principios para la gobernanza de nuevas tecnologías en un abordaje de abajo hacia arriba que permita cerrar dicha brecha.

4.1 Gobernanza EDI y datos

El objetivo de la gobernanza respecto de los datos es clave para un EDI. Éste se centra fundamentalmente en decidir cómo hacer para *compartir datos desde las fuentes auténticas, con seguridad*. Es la gobernanza la que debe establecer los protocolos que deben ser utilizados para compartir datos por todos los miembros en el espacio común, con miras a: a- evitar el acceso a los datos por personas o sistemas no autorizados; b- minimizar la desactualización de los datos que se comparten; c- minimizar la cantidad de errores en los datos y d- minimizar el tiempo para encontrar o identificar los datos.

Cabe señalar también, que las decisiones respecto de datos en un contexto EDI deben basarse en la aplicación del principio *once only* (una sola vez), el cual ha sido recogido por la Norma IRAM 17610-1 y refuerza el concepto de gestión de datos por sus fuentes auténticas.

Varias tecnologías de creciente adopción presentan potencial para un EDI, a la vez que desafíos en cuanto a su uso apropiado, especialmente más allá de los límites de una organización. Las mismas se apalancan en dos elementos comunes de interés: 1) alta disponibilidad de datos y 2) alta conectividad. De estos dos, el último resulta habilitante del primero, a punto que no cabe casi concebir aquél sin éste. Aún así, el rol transformador de los datos es innegable, llegando a representar un recurso estratégico, y por ende de ineludible relevancia a nivel de gobernanza. En el contexto de un EDI, los datos son insumo utilizable por su dimensión organizacional a partir del intercambio realizado entre los nodos participantes. Esta circunstancia revela que las dificultades o inexistencia de gobernanza de los datos a nivel intraorganizacional deberían ser resueltas en forma previa a la incorporación a un EDI. El hacerlo así asegura la disponibilización y visibilización de piezas clave para uso de los miembros y sienta buenas bases para el gobierno interorganizacional de los datos.

La acumulación de datos, tanto en forma estructurada como no estructurada, aunada al empleo de analítica de datos (frecuentemente auxiliada por el aprendizaje computacional), permite generar nuevo conocimiento para una mejor toma de decisiones, al detectar tendencias y patrones no fácilmente apreciables. Por otro lado, tanto este volumen de datos, como su diversidad y su velocidad de acopio, extreman la necesidad de un gobierno de datos a lo largo de su ciclo de vida. Tal cosa sólo puede comprenderse dentro de una dinámica que permita en todo momento poder recuperar eficientemente lo que es valioso, a la vez que no incurrir en violaciones de los términos legales y regulatorios en cuanto al almacenamiento y uso de la información o su eliminación. Es un hecho que tanto el abaratamiento del almacenamiento, en sus diversas formas, físicas o virtuales, como la aparición de dispositivos capaces de generar y transmitir datos en forma inmediata a través de redes de distinto alcance,

tientan a las organizaciones a una mayor retención de los datos, a la vez que hacen más desafiante su gobierno. Los dispositivos IoT constituyen un ejemplo de tales capacidades y su información podría ser incorporada a un EDI a través de los subsistemas organizacionales que los gestionan. En este caso, sus responsables deben comprender que al margen de las consideraciones en torno a la gestión de datos a nivel intraorganizacional, por tratarse de sistemas ciberfísicos, se suman a los problemas de la seguridad informática los vinculados a la seguridad física, su confiabilidad y su resiliencia [Smallwood, 2020]. Se trata de aspectos que en sentido estricto exceden a la gobernanza EDI, pero que sería beneficioso que ésta visibilizara a través de recomendaciones para sus miembros.

4.2 Gobernanza EDI y procesos

Respecto de los procesos, el objetivo de la gobernanza para un EDI es co-crear procesos intra e inter organizacionales/sistemas que permitan la coordinación de acciones para brindar servicios colectivamente y es aquí donde la gobernanza deberá decidir sobre los protocolos que se deben aplicar para permitir generar procesos proactivos, los cuales simplificarán la vida de las personas. Estas decisiones deben buscar: a- maximizar la reutilización funcional para los procedimientos comunes (generales, globales); b- maximizar la flexibilidad para soportar la diversidad de los casos particulares y locales de cada organización; c- maximizar la participación de las organizaciones involucradas en el diseño o rediseño de los procesos proactivos y d- maximizar la calidad y confidencialidad de los procesos.

Como se ha dejado ver, las decisiones en cuanto a protocolos/procesos deben basarse en la aplicación del principio procesos proactivos, asimismo plasmado en la Norma IRAM 17610-1, a fin de procurar una respuesta anticipada a las posibles necesidades.

Circunscripta al ámbito de una organización, la gobernanza de procesos define la forma en que se consolidan las iniciativas de gestión de sus procesos organizacionales, dentro de determinados estándares, normas y directrices que garantizan que los mismos persigan un objetivo común.

BPM (Business Process Management o Gestión de procesos de negocio) es una estrategia para gestionar y mejorar el rendimiento de un negocio optimizando sus procesos a través de la modelización, ejecución y medida de rendimiento dentro de un ciclo de mejora continua.

Los procesos y su gestión con la estrategia BPM aportan a una visión transversal de la organización, es decir, si bien las organizaciones se configuran en torno a áreas funcionales, los procesos que la organización ejecuta atraviesan dichas áreas, para lo cual es necesario reconocer las tareas que se realizan, el tiempo que demora cada una y la manera en que se estructuran.

En el contexto de un EDI, los procesos trascienden las fronteras organizacionales y también llevan a nivel interorganizacional la transversalidad que plantea la estrategia BPM clásica, aportando una trazabilidad de tareas y actividades entre organizaciones muy útil para la gobernanza del EDI, en torno a los procesos.

Los elementos centrales de BPM están definidos en [Kerpedzhiev et.al, 2021] como: 1-Alineación estratégica, que es la vinculación continua entre los procesos

organizacionales y los objetivos de negocio, 2- Gobernanza, que fija los procesos de rendición de cuentas y toma decisiones para guiar las acciones de negocio, 3 - Métodos que son los enfoques y técnicas que dan soporte y habilitan las acciones de los procesos y sus objetivos, 4- Tecnología, que la conforman el software, hardware y sistemas de información que dan soporte a las actividades de los procesos de negocio, 5 - Personas, que son individuos y grupos que mejoran y aplican, de manera continua, su experiencia y conocimiento y 6 – Cultura, que son los valores y creencias colectivos que dan forma a las actitudes y comportamientos relacionados con el proceso.

En relación a los métodos o marcos de trabajo de BPM existentes están siendo desafiados por cambios sociotécnicos como los provocados por la digitalización. En la adopción de nuevas tecnologías, la digitalización transforma los procesos existentes y permite nuevos debido a su impacto en el comportamiento y las necesidades individuales, la colaboración dentro y entre empresas y nuevas formas de automatización. Por ejemplo, las plataformas de colaboración social facilitan la formación de equipos que trabajan en procesos intensivos en conocimiento independientemente del tiempo y el lugar. La automatización de procesos robóticos y cognitivos permite la ejecución de tareas no estructuradas, mientras que Internet of Things y blockchain permiten procesos descentralizados [Kerpedzhiev et.al, 2021].

Por otra parte, se requiere también considerar avances tecnológicos respecto a la incorporación de nuevos modelos computacionales distribuidos y de aspectos sociales en la ejecución de los procesos de negocio y de las herramientas que la soportan. Por estos motivos, resultan relevantes las investigaciones en metodologías, marcos de trabajo y herramientas que incluyan estos nuevos paradigmas en la administración de procesos de negocio [Bazán P.,2015].

4.3 Gobernanza EDI y servicios

El objetivo de la gobernanza respecto de los servicios de un EDI es *liberar servicios abiertos y extensibles* para que otros desarrolladores puedan continuar combinando o integrando servicios de datos y procesos existentes. Esto extiende el concepto de software abierto a servicios abiertos a ser reutilizados, lo cual no debe entenderse como acceso abierto, dado que el control de acceso sigue residiendo en los proveedores de los datos o en los protocolos basados en eventos. Las decisiones deben buscar: a- maximizar la innovación permitiendo la integración o combinación de servicios de múltiples proveedores y diversas tecnologías; b- maximizar la extensibilidad de los servicios y aplicaciones por terceros, en cualquier tecnología; c- maximizar la flexibilidad para lograr servicios personalizados por diferentes canales.

Las decisiones relativas a servicios deben basarse en una interfaz abierta y estándar, buscando alcanzar la neutralidad tecnológica, al poder continuar un desarrollo existente con otras o más modernas tecnologías. La neutralidad tecnológica ha sido incorporada como principio a la Norma IRAM 17610-1.

La gobernanza del EDI para los servicios, debe lograr decidir, a través de un comité técnico, que combinación o integración de tecnologías es la más adecuada para satisfacer las necesidades del caso de uso que se esté analizando. Es aquí donde es posible obtener los mayores beneficios de contar con un EDI y también evita el caer en soluciones mono-tecnológicas que suelen generar obsolescencias generalizadas. La

selección de las tecnologías a partir de las necesidades y resultados esperados definidos es la clave de estas decisiones.

5. La gobernanza de un EDI y la transversalidad tecnológica

Dejando a un lado la existencia de productos fuertemente apoyados en las redes, como lo son los asociados a IoT, la alta conectividad en términos más generales ha permitido el florecimiento de un conjunto de funciones o prestaciones ofrecidas como servicio, abarcando desde la infraestructura hasta el nivel de aplicación. Tal modalidad, al permitir la contratación, ampliación o reducción de servicios bajo demanda, ha obtenido gran repercusión como facilitadora de la eficiencia en el uso de recursos TIC y gana creciente adopción. Desde el punto de vista de un EDI, de todos modos, la utilización de nubes públicas conlleva riesgos, que en general afectan al control sobre la información (incidentes de ciberseguridad, cohabitación de clientes, continuidad de negocios). Imagínese, por ejemplo, lo que sucedería si un servicio de nube es retirado del mercado y su proveedor no ofrece herramientas para copia de seguridad o migración. Cada órgano de gobernanza podrá optar por cualquier política particular para la implantación de nodos en nubes (públicas, privadas o híbridas), pero manteniendo a la vista las implicancias comentadas.

Hemos reservado las consideraciones relativas a la inteligencia artificial para este punto, ya que de todas las TIC en expansión, es la que reviste mayor potencial para un EDI, aunque también los más grandes riesgos. Esto es así, debido a su amplia transversalidad, con lo cual se combina versátilmente con cualquiera de las tecnologías previamente mencionadas.

Si bien la IA ha evolucionado a lo largo de décadas, es en los últimos años cuando se ha observado una mayor masificación de algunas de sus propuestas, principalmente gracias al impulso recibido por subdisciplinas como el aprendizaje computacional y aún más particularmente, el aprendizaje profundo. Entre los riesgos asociados a la utilización de estas técnicas, algunos de ellos aparecen más vinculados a la adecuación de las decisiones (e.g., sobresimplificación, sesgos) y otros más asociados a la responsabilidad por tales decisiones (e.g., transparencia, explicabilidad). Por este motivo, creemos oportuno señalar que en el caso del aprendizaje supervisado (e.g., procesamiento del lenguaje natural, reconocimiento de imágenes), el gobierno de datos cobra especial foco, ya que los utilizados para el entrenamiento impactan muy directamente en la corrección o incorrección de las decisiones tomadas por la IA. Naturalmente, la retroalimentación humana podría morigerar falencias en aquél sentido, pero no obra como sustituto. Por su parte, en el caso del aprendizaje por refuerzo y del aprendizaje no supervisado, los aspectos relacionados con el gobierno de procesos y algoritmos se vuelven más críticos.

El corolario para una gobernanza de la información y la tecnología en un contexto EDI se deriva inmediatamente: las aplicaciones de aprendizaje supervisado que se incorporen a través de sus miembros deben estar previamente gobernadas a nivel de datos en el ámbito intraorganizacional, lo cual supone, entre otras cosas, clasificar, priorizar y calendarizar la información para facilitar su gestión, y desarrollar las políticas, estándares y procesos conducentes a asegurar calidad en los datos. En cuanto

a los sistemas asociados a aprendizaje por refuerzo y aprendizaje no supervisado para similar contexto EDI, el prerrequisito aparece respecto del gobierno de procesos y algoritmos a nivel intraorganizacional, Cabe recordar aquí que, en el estado de evolución actual de la tecnología, las decisiones que más fácilmente pueden confiarse a una IA son las que corresponden a un nivel bajo a medio de complejidad, es decir, las que implican mayor acuerdo y menor incertidumbre [Hilb, 2020].

Las consideraciones que hemos detallado para la IA a nivel intraorganizacional pueden trasladarse entonces sin dificultad al nivel interorganizacional en donde la interacción de los bloques constructivos de datos, procesos y servicios, que utilicen o no IA, recibirá atención.

El concepto de RPA (*Robot Process Automation*) acuñado por IEEE Standards Association como: "el software que utiliza las reglas de negocio y la coreografía predefinida para completar la ejecución autónoma de una combinación de procesos, actividades, transacciones y tareas en uno o más sistemas de software no relacionados para entregar un resultado o servicio con gestión de excepción humana", es sin dudas una tecnología que aporta a que los protocolos de gobernanza de un EDI puedan cumplirse incluyendo desde los sistemas altamente interoperables hasta aquellos en los cuales se dificulta su modificación.

El RPA puede enfocarse hacia la automatización de tareas que reproducen el trabajo que hacen los humanos durante la ejecución de un proceso o hacia el uso de un robot de software clásico que ejecuta una tarea específica dentro del proceso. El primero se apoya en la arquitectura de TI mientras que el segundo se introduce en ella, de manera más intrusiva.

En este sentido, la madurez y características tecnológicas del nodo a integrar, marcarán a la gobernanza del EDI si utiliza un robot como instancia de software o como coreógrafo de las actividades interorganizacionales.

6. Conclusiones

Hemos reseñado a través de este trabajo algunas tecnologías de creciente adopción en las organizaciones, y con potencial impacto para un EDI. Por su relevancia para el funcionamiento de éste, así como para la visibilización de sus implicancias a nivel de gobernanza, se ha centrado la atención en el citado subconjunto, a la vez que se han omitido otras que resultan menos aplicables o necesarias, independientemente de su presencia actual en otros contextos.

Incidentalmente, las tecnologías consideradas pueden ser utilizadas para contribuir al gobierno de la información tanto como pueden ser objeto de gobierno ellas mismas, con lo cual su doble carácter se ve reforzado también desde esta mirada. Así, por ejemplo, la clasificación de datos a través del aprendizaje computacional puede facilitar su hallazgo, lo cual resulta de especial ayuda en el caso de los datos no estructurados. Herramientas como la IA resultan imprescindibles para mantener bajo control el acervo informativo a través del tiempo, particularmente a largo plazo. Similarmente, el desgobierno histórico de datos podría ser revertido eventualmente, merced a estas ayudas.

La gran cantidad de información utilizada por un EDI, en conjunción con las complejidades de la interacción entre organizaciones y la vertiginosa evolución tecnológica, han dado lugar al presente análisis, con el propósito de dar a conocer situaciones de diverso grado de evidencia, pero que necesariamente deben ser explicitadas para ser abordadas con el mejor criterio posible a nivel de gobernanza. A través de estas consideraciones se ha pretendido acercar elementos para la definición de principios en relación por las distintas gobernanzas EDI.

La definición de los mecanismos y órganos de decisión, así como los contenidos detallados de la gobernanza EDI continúan su desarrollo en el marco de un proyecto de investigación del que forman parte dos de los autores.

Referencias

1. Bazán, P. A., Luz Clara, H., Ceballos, J. L., Giorgetti, G., Ugalde, D. F., & Moreno, D. A. (2022). Integrabilidad y ecosistemas digitales: problemática, fundamentos y normalización. In *XXVIII Congreso Argentino de Ciencias de la Computación (CACIC) (La Rioja, 3 al 6 de octubre de 2022)*.
2. Hilb, M. (2020). Toward artificial governance? The role of artificial intelligence in shaping the future of corporate governance. Journal of Management and Governance. 24:851–870.
3. IRAM 17610-1. Ecosistema Digital de Integrabilidad (EDI) – Requisitos
4. ISO 24143:2022 Information and documentation — Information Governance — Concept and principles.
5. ISO 37000:2021- Governance of organizations — Guidance.
6. ISO/IEC 38500:2015 Information technology – Governance of IT for the organization.
7. ISO/IEC TR 38504:2016 Governance of information technology – Guidance for principles-based standards in the governance of information technology.
8. ISO/IEC 38505-1:2017 Part 1: Application of ISO/IEC 38500 to the governance of data.
9. ISO/TR 44000:2019 Principles for successful collaborative business relationship management.
10. Kavanagh, C. (2019). New Tech, New Threats, and New Governance Challenges: An Opportunity to Craft Smarter Responses? Carnegie Endowment for International Peace, Washington, DC, USA.
11. Kerpedzhiev, G. D., König, U. M., Röglinger, M., & Rosemann, M. (2021). An exploration into future business process management capabilities in view of digitalization: Results from a Delphi study. Business & Information Systems Engineering, 63, 83-96.
12. Smallwood, R. (2020). Information Governance. Concepts, Strategies and Best Practices, 2nd Ed. John Wiley & Sons, Inc., Hoboken, USA.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Contribución del proyecto CAP4CITY a los ODS 2030

Rocío Muñoz[0000-0001-6521-0568], Ariel Pasini[0000-0002-4752-7112], Patricia Pesado[0000-0003-0000-3482]

Instituto de Investigación en Informática LIDI (III-LIDI) *
Facultad de Informática – Universidad Nacional de La Plata 50 y 120 La Plata Buenos Aires
*Centro Asociado Comisión de Investigaciones Científicas de la Pcia. de Bs. As. (CIC)
{rmunoz,apasini,ppesado}@lidi.info.unlp.edu.ar

Abstract: Se presenta el conjunto de los ODS que conforman la Agenda 2030 impulsada por la Asamblea General de las Naciones Unidas y el proyecto CAP4CITY, cuyo objetivo principal es utilizar la gran atención que el concepto de SSC ha alcanzado en América Latina e integrarlo en diversos cursos universitarios, desarrollando nuevos planes de estudio en todos los niveles del proceso educativo. El artículo propone analizar cómo el proyecto, y los cursos desarrollados por el mismo, contribuyen a los ODS, vinculando los contenidos y objetivos con las metas y consideraciones que propone la Agenda 2030. Finalmente, se brindan algunos de los resultados obtenidos finalizados los 4 años de duración del proyecto CAP4CITY.

Keywords: Objetivos de Desarrollo Sostenible - ODS, Ciudades Inteligentes Sostenibles - SSC, Agenda 2030, Programas educativos en SSC.

1. Introducción

La llamada Agenda 2030 [1] de las Naciones Unidas contiene 17 Objetivos de Desarrollo Sostenible (ODS), los cuales aplican a todos los países, ricos y pobres, pudiendo cada uno de ellos impulsar diferentes metas, según sus prioridades. En total, estos ODS presentan un conjunto de 169 metas, las cuales van mucho más allá al incluir diferentes retos tales como el cambio climático, la desigualdad económica, el consumo sostenible, paz y justicia, entre otros.

El proyecto "Strengthening Governance Capacity for Smart Sustainable Cities (CAP4CITY)" [2] es un proyecto con una naturaleza interdisciplinaria, contando con expertos de diferentes áreas, tales como administración de empresas, informática, ingeniería, arquitectura y urbanismo, planificación urbana, ciencias políticas, entre otros, que buscan 3 objetivos concretos [3]: 1- mejorar la calidad de las herramientas, metodologías y enfoques pedagógicos de aprendizaje y enseñanza en la región de América Latina; 2- desarrollar contenidos didácticos y resultados de aprendizaje basados en las TIC para desarrollar competencias específicas en el área de Ciudades Inteligentes Sostenibles (SSC, por sus siglas en inglés); y 3- entregar contenido utilizando prácticas basadas en TIC a estudiantes de diferentes niveles educativos.

En el marco del proyecto, se llevaron a cabo diversas actividades con el fin de crear nuevos cursos universitarios que se vinculen con las áreas y las competencias necesarias para la concientización sobre SSC en la región. Así es como, con la participación de 360 actores locales de distintas disciplinas de Argentina, Brasil, Chile y Colombia, se identificaron 93 competencias a abordar, que dieron paso a la creación de 31 cursos

divididos en 9 áreas de conocimiento [4]: 1- Comunicación; 2- Gobernanza; 3- Sistemas de Información; 4- Implicaciones Legales y Éticas de las SSC; 5- Administración Pública y Gestión; 6- Negocios y economía; 7- Fundamentos de la Ciudad Inteligente; 8- TIC Socio-Técnico; y 9- Estudios Urbanos y Sostenibilidad.

Tanto el proyecto CAP4CITY en general, como cada uno de los 31 cursos, ofrecen una contribución a uno o más ODS. En particular, el proyecto se encuentra estrechamente relacionado con el ODS 4- Educación de calidad, y los cursos, en su gran mayoría, con los ODS 9- Industria, innovación e infraestructura, 11- Ciudades y comunidades sostenibles y 17- Alianzas para lograr los objetivos. El resto de los ODS también se encuentran vinculados con entre 2 y 8 cursos cada uno, a excepción de los ODS 2- Hambre cero y 5- Igualdad de género, que no tienen una relación directa con ninguno de estos.

En la sección dos se presenta la Agenda 2030 de las Naciones Unidas con los ODS. En la tercera sección se da una introducción al proyecto CAP4CITY y se definen los objetivos y la definición de los cursos que el mismo ofrece. Luego, en la sección 4, se establece una relación entre los cursos y su contribución a los ODS teniendo en cuenta ciertos puntos a considerar. La quinta sección brinda los aportes más relevantes del proyecto y la importancia que tiene la incorporación de los cursos relacionados con las SSC en los programas educativos. Por último, la sección 6 presenta las conclusiones de este artículo.

2. Objetivos de Desarrollo Sostenible

En el año 2015, la Asamblea General de las Naciones Unidas [5] y los líderes mundiales adoptaron un conjunto de objetivos globales con el fin de erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos, lanzando así una nueva agenda de desarrollo sostenible.

La llamada Agenda 2030 [1] contiene 17 ODS, los cuales aplican a todos los países, ricos y pobres, pudiendo cada uno de ellos impulsar diferentes metas, según sus prioridades.

En total, estos ODS presentan un conjunto de 169 metas y completan la agenda de los pasados ocho Objetivos de Desarrollo del Milenio (ODM) [6], culminada en el año 2015. Estos nuevos objetivos y metas van mucho más allá al incluir diferentes retos tales como el cambio climático, la desigualdad económica, el consumo sostenible, paz y justicia, entre otros.

Los 17 objetivos de la Agenda 2030 se centran en diferentes aspectos de lo que entendemos como sostenibilidad. Así, cada uno de estos objetivos hace hincapié en cuestiones de suma importancia para todos los países.

- ODS 1 – Fin de la pobreza: Poner fin a la pobreza en todas sus formas en todo el mundo
- ODS 2 – Hambre cero: Poner fin al hambre, lograr la seguridad alimentaria y la mejora de la nutrición y promover la agricultura sostenible
- ODS 3 – Salud y bienestar: Garantizar una vida sana y promover el bienestar para todos en todas las edades

- ODS 4 – Educación de calidad: Garantizar una educación inclusiva, equitativa y de calidad y promover oportunidades de aprendizaje durante toda la vida para todos
- ODS 5 – Igualdad de género: Lograr la igualdad de géneros y empoderar a todas las mujeres y niñas
- ODS 6 – Agua limpia y saneamiento: Garantizar la disponibilidad de agua y su gestión sostenible y el saneamiento para todos
- ODS 7 – Energía asequible y no contaminante: Garantizar el acceso a una energía asequible, segura, sostenible y moderna para todos
- ODS 8 – Trabajo decente y crecimiento económico: Promover el crecimiento económico sostenido, inclusivo y sostenible; el empleo pleno y productivo y el trabajo decente para todos
- ODS 9 – Industria, innovación e infraestructura: Construir infraestructuras resilientes, promover la industrialización inclusiva y sostenible y fomentar la innovación
- ODS 10 – Reducción de las desigualdades: Reducir las desigualdades en y entre países
- ODS 11 – Ciudades y comunidades sostenibles: Lograr que las ciudades y los asentamientos humanos sean inclusivos, seguros, resilientes y sostenibles
- ODS 12 – Producción y consumo responsables: Garantizar modalidades de consumo y producción sostenibles
- ODS 13 – Acción por el clima: Adoptar medidas urgentes para combatir el cambio climático y sus efectos
- ODS 14 – Vida submarina: Conservar y utilizar en forma sostenible los océanos, los mares y los recursos marinos para el desarrollo sostenible
- ODS 15 – Vida de ecosistemas terrestres: Promover el uso sostenible de los ecosistemas terrestres, luchar contra la desertificación, detener e invertir la degradación de las tierras y frenar la pérdida de la diversidad biológica
- ODS 16 – Paz, justicia e instituciones sólidas: Promover sociedades pacíficas e inclusivas para el desarrollo sostenible, facilitar el acceso a la justicia para todos y crear instituciones eficaces, responsables e inclusivas a todos los niveles
- ODS 17 – Alianzas para lograr los objetivos: Fortalecer los medios de ejecución y revitalizar la alianza mundial para el desarrollo sostenible

3. Proyecto CAP4CITY

El proyecto “Strengthening Governance Capacity for Smart Sustainable Cities (CAP4CITY)” [2] es un proyecto financiado como parte del programa Erasmus+ de la Unión Europea, que fue ejecutado entre los años 2018 y 2022 por un consorcio integrado por 12 universidades de América Latina y Europa. En total participaron 8 países: Austria, Estonia, Países Bajos, Polonia, Brasil, Chile, Colombia y Argentina.

El proyecto tuvo una naturaleza interdisciplinaria, contando con expertos de diferentes áreas, tales como administración de empresas, informática, ingeniería, arquitectura y urbanismo, planificación urbana, ciencias políticas, entre otros.

El compromiso del proyecto CAP4CITY es capacitar profesionales de diferentes ramas para que sean líderes profesionales, buscando facilitar la vida de los ciudadanos con el impulso de la economía, la educación, la movilidad y las nuevas empresas de tecnología y negocios que sean garantía de Ciudades Inteligentes Sostenibles.

3.1 Objetivo

El fin del proyecto CAP4CITY es utilizar la gran atención que el concepto de SSC ha alcanzado en América Latina e integrarlo en diversos cursos universitarios, utilizando nuevas e innovadoras metodologías, herramientas de enseñanza y de aprendizaje, desarrollando nuevos planes de estudio en todos los niveles del proceso educativo. El objetivo final es fortalecer las capacidades de gobernanza para facilitar la implementación de iniciativas de ciudades inteligentes.

Se define una SSC [3] como una ciudad innovadora que utiliza Tecnologías de Información y Comunicación (TIC) y otros medios para mejorar la calidad de vida, la eficiencia de la operación y los servicios urbanos, y la competitividad, al tiempo que se garantiza que satisfaga las necesidades de las generaciones presentes y futuras con respecto a los aspectos económicos, sociales, ambientales y culturales. En particular, el proyecto busca:

- Mejorar la calidad de las herramientas, metodologías y enfoques pedagógicos de aprendizaje y enseñanza en la región de América Latina.
- Desarrollar contenidos didácticos y resultados de aprendizaje basados en las TIC para desarrollar competencias específicas en el área de SSC.
- Entregar contenido utilizando prácticas basadas en TIC a estudiantes de diferentes niveles educativos.

3.2 Cursos

El primer paso para la definición de los nuevos cursos a crear en el marco del proyecto fue la evaluación del estado del arte de los programas educativos relacionados con las SSC en la región. Para ello se evaluaron los planes de estudio de las diferentes carreras acreditadas en los países de América Latina, teniendo en cuenta si involucraban conceptos relacionados con la definición del proyecto de una SSC.

Con el fin de sensibilizar sobre el estado del arte, y en busca de identificar las competencias necesarias para el proyecto y proponer modelos pedagógicos, el siguiente paso fue la realización de talleres en los cuatro países de América Latina [7].

Estos talleres contaron con la participación de actores locales, permitiendo que el proyecto incorpore las necesidades específicas de la región. La tarea de los participantes fue formar grupos e identificar áreas y, luego, habilidades en relación a esas áreas, con el fin de identificar las competencias y los modelos pedagógicos de acuerdo a éstas. Luego de las discusiones grupales, los participantes presentaron y discutieron en sesiones plenarias las conclusiones de cada grupo.

En total participaron, entre los talleres realizados en Argentina, Brasil, Chile y Colombia, 360 actores locales de distintas disciplinas, de varias áreas tales como educación, gubernamentales, no-gubernamentales y empresas privadas.

Finalmente, el proyecto desarrolló una biblioteca de 31 cursos universitarios, basados en las 93 competencias identificadas a través de los talleres y una amplia evaluación comparativa internacional sobre las capacidades necesarias para los profesionales de las SSC. Los 31 cursos se dividen en 9 áreas de conocimiento, agrupadas según capacidades y temática, como se puede observar en la Tabla 1.

Tabla 1. Cursos desarrollados por el proyecto CAP4CITY

1- Comunicación	**6- Negocios y economía**
Comunicación e interacción en redes sociales	Principios de Sostenibilidad Económica
2- Gobernanza	Innovación para la Transformación Urbana
E-Democracia y E-Participación	**7- Fundamentos de la Ciudad Inteligente**
Gobernanza Basada en la Evidencia y en la Colaboración	Aplicaciones de Ciudad Inteligente
Gobernanza de Ciudades Inteligentes	Ciudad Inteligente: Contexto, Política y Gobierno
Diseño de Servicios Impulsado por el Usuario y Co-Creación	Introducción a las Ciudades Inteligentes y Sostenibles
3- Sistemas de Información	**8- TIC Socio-Técnico**
Gobierno de Datos y Gestión de la Información	Análisis de Datos para SSC, Toma de Decisiones
Sist. de Información para Apoyo a la Toma de Decisiones	Basada en Datos
4- Implicaciones Legales y Éticas de las SSC	Infraestructura Digital y Urbana
Ciberseguridad, Privacidad e Implicaciones Éticas	Tecnologías Disruptivas en las SSC
Competencias Legales para las SSC	Desarrollo de TI
5- Administración Pública y Gestión	Gobernanza y Gestión de TI
Estructuras y Procesos Org. a Nivel Local	Gobierno Abierto
Gestión y Estrategias para la Transformación Urbana	**9- Estudios Urbanos y Sostenibilidad**
Métodos y Herramientas para la Gestión de la Ciudad	Sostenibilidad Económica
Monitoreo y Evaluación de Soluciones	Sostenibilidad Ambiental
Project Management para Proyectos de SSC	Sostenibilidad Social
Planificación de Proyectos Relacionados con las SSC	Arquitectura Sostenible
	Planificación Urbana

3.3 Validación de los cursos

Luego de definidos los cursos por el proyecto CAP4CITY se llevó a cabo una validación de los mismos con el fin de ayudar a definir el modo de implementación más adecuado para cada curso.

Los miembros del proyecto de los países de América Latina realizaron una recopilación de información [8] mediante un formulario, realizando entrevistas en línea, encuestas, encuentros presenciales o utilizando programas ya existentes. Dicho formulario fue repartido entre un gran grupo de expertos en el área para evaluar la forma en que se debería implementar cada curso definido por el CAP4CITY: pregrado, posgrado y/o curso masivo abierto y en línea (Massive Online Open Course, conocido como MOOC) para todo público. A cada interesado se le brindó un documento con la información de cada uno de los cursos, los objetivos de aprendizaje, de cuántas clases constaría y una descripción de los contenidos del mismo.

Se obtuvo así una totalidad de 153 opiniones, entre expertos del gobierno, del área educativa y de sectores privados, acerca de los cursos definidos. Además, se realizó una evaluación externa por parte de 11 expertos internacionales en el área de las SSC.

4. CAP4CITY y su contribución a los ODS

En términos generales, al evaluar los objetivos que propone el proyecto CAP4CITY, se observa una estrecha relación con el Objetivo de Desarrollo Sostenible 4 – Educación de calidad.

De este ODS 4 se proponen varias metas [9] las cuales se relacionan con el proyecto en cuanto a que busca asegurar el acceso igualitario para todos a una formación técnica, profesional y de calidad, incluida la enseñanza universitaria, y aumentar el número de personas con competencias para acceder al empleo y el trabajo decente.

4.1 Relevamiento de los cursos del proyecto CAP4CITY

De cada uno de los 31 cursos desarrollados por el proyecto CAP4CITY, se analizaron los contenidos mínimos y los objetivos de aprendizaje propuestos [4] con el fin de encontrar a cuáles de cada uno de los ODS se brinda una contribución.

La Tabla 2 muestra los resultados del análisis de la contribución de los cursos del proyecto CAP4CITY a los ODS.

Tabla 2. Contribución de los cursos a los ODS

	1	3	6	7	8	9	10	11	12	13	14	15	16	17
Comunicación e interacción en redes sociales						X								X
E-Democracia y E-Participación	X					X	X	X					X	
Gobernanza Basada en la Evidencia y en la Colaboración														X
Gobernanza de Ciudades Inteligentes								X	X					X
Diseño de Servicios Impulsado por el Usuario y Co-Creación	X	X	X	X				X	X		X			
Gobierno de Datos y Gestión de la Información					X								X	X
Sistemas de Información para Apoyo a la Toma de Decisiones						X								
Ciberseguridad, Privacidad e Implicaciones Éticas						X							X	X
Competencias Legales para las SSC													X	X
Estructuras y Procesos Organizacionales a Nivel Local						X								
Gestión y Estrategias para la Transformación Urbana					X	X				X				X
Métodos y Herramientas para la Gestión de la Ciudad						X		X						
Monitoreo y Evaluación de Soluciones								X						
Project Management para Proyectos de SSC						X								X

Planificación de Proyectos Relacionados con las SSC	X															X
Principios de Sostenibilidad Económica	X				X			X								
Innovación para la Transformación Urbana						X										X
Aplicaciones de Ciudad Inteligente		X		X		X		X	X							
Ciudad Inteligente: Contexto, Política y Gobierno					X	X		X								
Introducción a las Ciudades Inteligentes y Sostenibles	X	X			X	X		X	X							
Análisis de Datos para SSC, Toma de Decisiones Basada en Datos																X
Infraestructura Digital y Urbana						X		X								X
Tecnologías Disruptivas en las SSC						X		X								X
Desarrollo de TI						X										
Gobernanza y Gestión de TI						X		X								X
Gobierno Abierto															X	X
Sostenibilidad Económica	X				X	X	X									X
Sostenibilidad Ambiental			X	X				X	X	X	X	X				
Sostenibilidad Social	X					X	X									
Arquitectura Sostenible						X		X								
Planificación Urbana	X					X		X					X	X		

Al analizar los datos relevados, se puede concluir que una gran cantidad de los cursos desarrollados se vinculan de forma directa con los ODS 9, 11 y 17. Los ODS 2, 4 y 5 no tienen una relación estrecha con ninguno de los 31 cursos, aunque se debe tener en cuenta que el ODS 4, como se mencionó anteriormente, está vinculado al proyecto en general, más allá de los contenidos impartidos por los cursos. El resto de los ODS se encuentran vinculados con entre 2 y 8 cursos cada uno. Para ilustrar mejor estas conclusiones, la Figura 1 muestra en forma de gráfico estos datos.

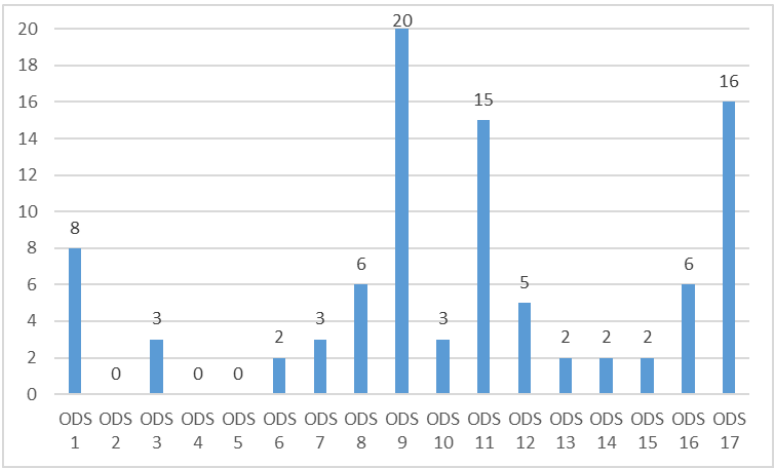


Figura 1. Cantidad de cursos que contribuyen a los ODS

4.2 Consideraciones para el relevamiento

Los datos mostrados en la tabla y gráfico de la sección anterior dan una idea de la importancia de los cursos desarrollados en el proyecto CAP4CITY en materia de concientización sobre los Objetivos de Desarrollo Sostenible.

Al igual que para vincular el objetivo del proyecto en general con el ODS 4 se tuvieron en cuenta las metas propuestas, para poder establecer la relación de cada uno de los 31 cursos con los ODS se toman en consideración las siguientes cuestiones [9]:

- ODS 1: Se debe garantizar que todas las personas tengan los mismos derechos a los recursos económicos y acceso a los servicios básicos, que el crecimiento económico sea inclusivo para crear empleos y promover la igualdad, y que se implementen sistemas y medidas de protección social.
- ODS 3: Se debe lograr la cobertura sanitaria universal, el acceso a servicios de salud y medicamentos eficaces, asequibles y de calidad para todos.
- ODS 6: Se busca lograr el acceso a servicios de saneamiento e higiene adecuados para todas las personas y reducir la contaminación, eliminando y minimizando la emisión de productos químicos, aumentando el reciclado, tratamiento y reutilización de aguas residuales.
- ODS 7: Se debe asegurar el acceso universal a los servicios de energía modernos, mejorar el rendimiento y aumentar el uso de fuentes renovables.
- ODS 8: Se tienen que asegurar las condiciones necesarias para que las personas accedan a empleos de calidad, estimulando la economía sin dañar al medio ambiente, y se deben lograr niveles más elevados de productividad económica mediante la modernización tecnológica y la innovación.
- ODS 9: Se busca aumentar la investigación científica y mejorar la capacidad tecnológica de los sectores industriales, fomentando la innovación. Las inversiones en infraestructura (transporte, riego, energía y tecnología de la información y las comunicaciones) resultan fundamentales y permiten a las ciudades ser más resistentes al cambio climático, impulsar el crecimiento económico y la estabilidad social.
- ODS 10: Se cree que el crecimiento económico no es suficiente para reducir la pobreza si este no es inclusivo y no tiene en cuenta las tres dimensiones del desarrollo sostenible: económica, social y ambiental. Se busca potenciar y promover la inclusión social, económica y política de todas las personas.
- ODS 11: Se desea crear ciudades con mayores oportunidades en cuanto a acceso a servicios básicos, energía, vivienda, transporte público, etc. para todas las personas, buscando aprovechar mejor los recursos y reduciendo la contaminación, la pobreza y el impacto ambiental negativo per cápita.
- ODS 12: Se busca hacer más y mejores cosas con menos recursos, incrementando las ganancias de bienestar de las actividades económicas mediante la reducción de la utilización de los recursos, la degradación y la contaminación durante todo el ciclo de vida. Se trabaja en conjunto entre varios actores de distintos ámbitos para lograrlo.
- ODS 13: Se desea mejorar la educación, la sensibilización y la capacidad humana e institucional para mitigar el cambio climático, adaptarse a él, alertar tempranamente y reducir los efectos.
- ODS 14: Se requiere una gestión prudente de los océanos y mares, cauces vitales

del comercio y el transporte, para conseguir un futuro sostenible.

- ODS 15: Se busca asegurar la conservación y el uso sostenible de los ecosistemas terrestres, de agua dulce, los bosques, humedales, montañas, etc. con las obligaciones contraídas mediante acuerdos internacionales
- ODS 16: Se desea ampliar y fortalecer la participación de los países en desarrollo en las instituciones de gobernanza mundial, creando instituciones eficaces y transparentes y garantizando el acceso a la información, conforme a las leyes nacionales y los acuerdos internacionales.
- ODS 17: Se necesitan alianzas entre los gobiernos, el sector privado y la sociedad civil, que sean inclusivas y se construyan sobre la base de principios y valores que otorgan prioridad a las personas y al planeta. Se busca mejorar la cooperación en materia de ciencia, tecnología e innovación, aumentar el uso de las tecnologías de la información y las comunicaciones y aprovecha las iniciativas existentes para elaborar indicadores basados en datos.

5. Diseminación de contenidos

Con los contenidos de los cursos desarrollados por el proyecto CAP4CITY se implementaron un total de 36 cursos universitarios, algunos de ellos completamente nuevos, y otros ya existentes que adaptaron su contenido. Algunas universidades de América Latina innovaron creando cursos cortos de uno o dos días, aumentando así el número de participantes alcanzados. Según los datos relevados a diciembre de 2022, 32 de estos 36 cursos fueron dictados y un total de 558 alumnos fueron alcanzados: 192 estudiantes de cursos cortos, 19 de especializaciones, 119 de pregrado y 228 de maestría y doctorado [10].

Por otro lado, se creó un MOOC de 6 semanas con videos de los diferentes contenidos de los cursos del proyecto CAP4CITY, el cual se brindó a través de la plataforma edX [11] y contó con un total de 1147 alumnos matriculados de 57 países que lo llevaron a cabo.

La incorporación de los cursos relacionados con SSC en los programas educativos creó un efecto secundario positivo en términos de compromiso de los estudiantes con el tema y en la importancia de la sostenibilidad, en particular de los ODS, generando además la voluntad de continuar la educación inscribiéndose en otros cursos ofrecidos por el proyecto.

6. Conclusiones

Como forma de ampliar y completar los ODM, la Asamblea General de las Naciones Unidas presenta un conjunto de 17 ODS, los cuales incorporan retos tales como el cambio climático, la desigualdad económica, el consumo sostenible, paz y justicia, entre otros, y se aplican a todos los países, ricos y pobres.

Con el fin de sensibilizar acerca de la importancia de las SSC, se presenta el proyecto CAP4CITY cuyo objetivo principal es integrar estos conceptos de sostenibilidad en diversos cursos universitarios de varias áreas, desarrollando nuevos

planes de estudio en todos los niveles del proceso educativo. Se presentan así el conjunto de 31 cursos creados y validados por parte del proyecto, divididos en 9 áreas de conocimiento y basados en un relevamiento de 93 competencias a través de talleres con actores locales.

Se presentó un análisis de cómo el proyecto CAP4CITY, y los cursos desarrollados por el mismo, contribuyen a los ODS, vinculando los contenidos mínimos de estos y los objetivos de aprendizaje propuestos con las metas y consideraciones que propone la Agenda 2030. Este análisis permitió demostrar cómo la incorporación de cursos relacionados con SSC en los programas educativos contribuyeron a la concientización sobre la importancia de los ODS, habiendo alcanzado a más de 1700 estudiantes de distintos niveles y regiones.

Referencias

[1] ONU, "ODS – AGENDA 2030" 2015. [Online]. Available: https://www.un.org/sustainabledevelopment/es/development-agenda/ [Accessed: 08Aug-2023].

[2] CAP4CITY-Project, "Strengthening Governance Capacity for Smart Sustainable Cities-CAP4CITY" 2018. [Online]. Available: https://www.cap4city.eu/home/ [Accessed: 08-Aug-2023].

[3] De Giusti, P. Pesado, A. Pasini, P. Thomas, R. Muñoz, J. S. Preisegger, K. Cenci, C. Chesñevar, E. Estevez, P. Fillottrani, A. S. Sergio Gómez, S. Rueda, "Fortalecimiento de las capacidades de gobernanza para ciudades inteligentes sostenibles" *WICC 2019*, 2019.

[4] CAP4CITY-Project, "La biblioteca de Cursos CAP4CITY Para Desarrollar Capacidades en Ciudades Inteligentes y Sostenibles" 2023.

[5] ONU, "Organización de las Naciones Unidas" [Online]. Available: https://www.un.org/es/ [Accessed: 08-Aug-2023].

[6] CEPAL, "Obejtivos de Daserrollo del Milenio" 2015. [Online]. Available: https://www.cepal.org/es/temas/objetivos-de-desarrollo-del-milenio-odm/objetivosdesarrollo-milenio [Accessed: 08-Aug-2023].

[7] CAP4CITY-Project, "Report on the workshops" 2020.

[8] CAP4CITY-Project, "Report– Validation Workshops Methodology" 2021.

[9] Consejo Nacional de Coordinación de Políticas Sociales, "Agenda 2030 -Informe de País 2021 Seguimiento de los progresos hacia las metas de los 17 ODS" 2021.

[10] CAP4CITY-Project, "Final Technical Report" 2023.

[11] CAP4CITY-Project, "Smart and Sustainable Cities: New Ways of Digitalization & Governance" *edX*, 2022. [Online]. Available: https://learning.edx.org/course/coursev1:DelftX+SaSC01x+2T2022/home [Accessed: 08-Aug-2e023].

Coordinadores:
- Armando De Giusti (UNLP)
- Marcelo Estayno (UNSAM)
- Mónica Tugnarelli (UNER)

CACiC 2023

CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Track de

Short Papers Alumnos

SPA

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Hacia el aprendizaje de la Revolución de Mayo en nivel primario mediante un Chatbot de Lenguaje Natural

Sergio Orciari[1], Marisa Panizzi[1], Iris Sattolo[1]

[1]Escuela Superior de Ingeniería, Informática y Ciencias Agroalimentarias Universidad de Morón. Cabildo 134 (B1708JPD), Partido de Morón, Argentina.
sergio.orciari@gmail.com, marisapanizzi@outlook.com, iris.sattolo@gmail.com

Resumen. En los tiempos que corren, la tecnología es un pilar fundamental para la sociedad y la educación dándoles una gran herramienta. El presente trabajo de tesis de la carrera Licenciatura de Sistemas de la UM pretende desarrollar un Chatbot de Lenguaje Natural que permita mejorar la experiencia de aprendizaje, ayudar a los estudiantes a aprender a su propio ritmo, aumentar la accesibilidad, crear una herramienta educativa escalable y fomentar el aprendizaje colaborativo. Antes de comenzar con la construcción del Chatbot de Lenguaje Natural, se construyó el estado del arte mediante un mapeo sistemático de la literatura (en inglés, *systematic mapping study* o SMS). Se presentan los resultados logrados mediante el desarrollo del SMS y se describen las actividades definidas para la finalización del trabajo de tesis.

Palabras claves: chatbot, educación, lenguaje natural, machine learning, nivel primario, revolución de mayo

1. Introducción

Luego de pasar un acontecimiento global e histórico cómo la Pandemia del 2020 y 2021, nos enfrentamos a partir de ahora a un cambio social y educativo mucho más profundo de lo que posiblemente podíamos llegar a imaginar.

Tenemos la oportunidad de generar nuevas respuestas a estas nuevas necesidades, y no solo de hacerlo, sino de hacerlo bien.

Todo parece indicar que el mundo online seguirá adquiriendo el protagonismo que merece, entendiéndolo como un entorno que convive con la presencialidad educativa, y que lejos de lo esperado nos permite interactuar y socializar más de lo que se creía, que nos permite emocionarnos con lo que hacemos y sentirnos parte de un todo, mucho más grande de lo que pensábamos.

Pero debemos tener presente que la educación online no consiste simplemente en adaptar de forma improvisada el contenido educativo a plataformas online para que los alumnos puedan enviar sus tareas y exámenes, dar clases virtualmente o establecer una vía de comunicación.

La docencia online requiere de un conjunto de recursos para asegurar que el estudiante esté acompañado en todo su proceso de aprendizaje, que se cuenta con los apoyos adecuados y con la experiencia y preparación necesarias para ofrecer recursos de calidad, trabajo en equipo entre docentes y estudiantes y, cómo no, un sólido modelo educativo y pedagógico.

Con el objetivo de poder ayudar a los estudiantes y docentes se planteó generar un Chatbot de Lenguaje Natural para responder preguntas sobre la Revolución de Mayo en Argentina [1].

Con el fin de conocer el estado del arte con respecto al uso de los Chatbot de Lenguaje Natural, en el dominio educativo, es que se ha realizado un mapeo sistemático de la literatura de acuerdo con los directrices propuestas por Kitchenham *et al.* [2].

El artículo se estructura de la siguiente manera: en la Sección 2 se describe la planificación del SMS, en la Sección 3 se describe su ejecución y los resultados. En la Sección 4 se exponen las conclusiones y trabajos futuros.

2. Planificación del SMS

Se presenta la definición del protocolo de revisión del SMS: preguntas de investigación (PI), estrategia de búsqueda, selección de los estudios, criterios y proceso de selección, formulario de extracción y el proceso de síntesis de los datos.

El objetivo de este SMS es responder la siguiente pregunta de investigación (PI): *¿Cómo se puede ayudar mediante la tecnología a los alumnos de colegios primario/secundario en la Argentina a adquirir conocimientos de la Revolución de Mayo?*

Esta pregunta principal se descompone en un conjunto de sub-preguntas (PI1-5), las cuales se presentan en la Tabla 1 junto con su motivación.

Tabla 1. Preguntas de investigación (PI) y su motivación.

Pregunta de investigación (PI)	Motivación
PI1: ¿Cómo se comunican los Chatbots con los usuarios?	Descubrir de qué forma los Chatbots se comunican con los usuarios.
P2: ¿Cuál es la finalidad/utilidad de los Chatbots?	Identificar cuál es el uso que los usuarios le dan a los Chatbots.
PI3: ¿Con qué tecnología están construidos los algoritmos de lenguaje natural de los Chatbots?	Identificar las tecnologías de programación utilizadas para la inteligencia artificial de lenguaje natural.
PI4: ¿Cuáles son los distintos tipos de interfaz del usuario que utilizan los Chatbots?	Agrupar los distintos tipos de interfaz que los Chatbots pueden utilizar para comunicarse con el usuario.
PI5: ¿Qué tipos de investigación se encuentra en los artículos?	Identificar los tipos de investigación realizada en los estudios primarios de acuerdo con la clasificación propuesta por Wieringa *et al*. [3].

Se definieron para la búsqueda de artículos las siguientes librerías, plataformas y repositorios digitales: *IEEE Xplore, Springer y Google Schoolar*, considerando artículos de congresos y artículos de revistas. El período comprendido definido ha sido entre enero 2010 hasta agosto de 2022. La cadena de búsqueda resultante es:

(chatbot educación) OR (chatbot lenguaje natural) OR (chatbot lenguaje natural tensorflow) OR (education chatbot) OR (chatbot school) OR (natural language processing)

En la Tabla 2, se presentan los criterios de inclusión y exclusión utilizados para el proceso de selección de artículos.

Tabla 2. Criterios de inclusión y exclusión.

Criterios de inclusión	Criterios de exclusion
Artículos que contengan los términos de búsqueda en el título, palabras clave, resumen, o introducción.	Artículos a los que no se tenga acceso.
Artículos duplicados. Cuando se encuentren varios artículos escritos por los mismos autores sobre el mismo tema o similar, se seleccionará el más completo.	Artículos que sólo se tenga acceso al resumen, presentaciones de PowerPoint, tesis doctorales, libros y literatura gris.
Artículos en idioma español e inglés	

El proceso de selección de los estudios consistió en realizar la búsqueda en las fuentes definidas aplicando la cadena en el título y/o en el resumen, para luego eliminar los artículos duplicados y aplicar los criterios de inclusión y exclusión en el título, resumen y palabras clave, después se aplicaron los criterios de inclusión y exclusión al texto completo.

Para dar respuesta a cada una de las preguntas de investigación (PI) se definió un esquema de clasificación que junto con el formulario de extracción de datos se presenta en un apéndice por restricciones de espacio [4].

3. Ejecución y resultados del SMS

Se encontraron 26 artículos de los cuales se analizaron 12 estudios primarios que se encuentran en el apéndice [4]. Los resultados del SMS para dar respuesta a las preguntas de investigación en base a la literatura analizada se presentan mediante gráficos.

PI1: ¿Cómo se comunican los Chatbots con los usuarios?

Según el tipo de interacción existen los siguientes tipos de Chatbots: textuales, multimedia y de voz. De la lectura de los documentos se obtiene como resultado que la mayor cantidad son de tipo de comunicación a través del texto. (Ver Figura 1).

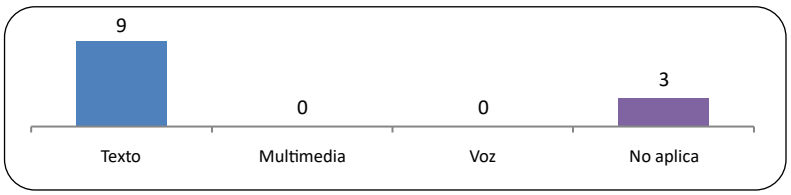


Figura. 1. Dimensión Tipo de Comunicación.

P2: ¿Cuál es la finalidad/utilidad de los Chatbots?

Las principales ramas en las cuales los artículos se aplican son educación, soporte de usuario y otros. (Ver Figura 2).

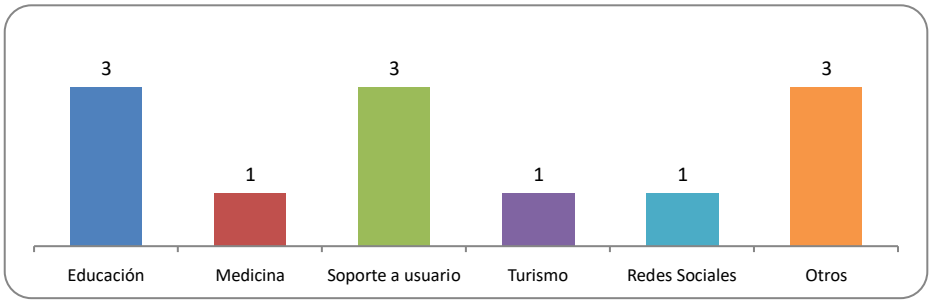


Figura. 2. Dimensión Finalidad.

PI3: ¿Con qué tecnología están construidos los algoritmos de lenguaje natural de los Chatbots?

Las principales tecnologías comerciales que podemos encontrar en los artículos son TensorFlow, Dialogflow y otras (desarrollados a medida) (Ver figura 3).

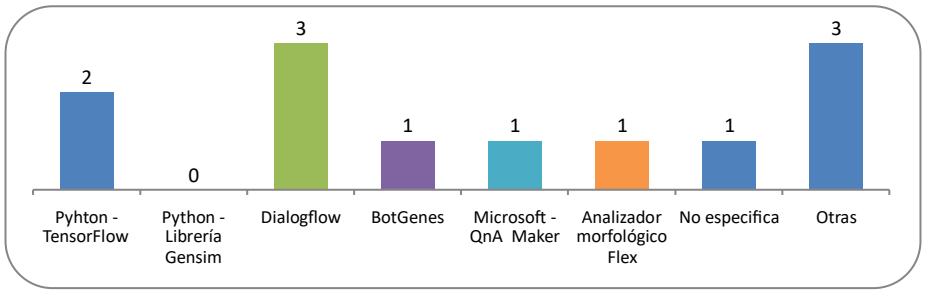


Figura. 3. Dimensión Tecnología.

PI4: ¿Cuáles son los distintos tipos de interfaz del usuario que utilizan los Chatbots? Se logro evidenciar que la interfaz principal para que el Chatbot se comunique es la Web. (Ver Figura 4).

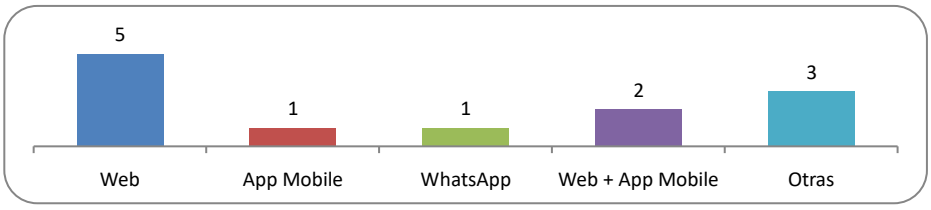


Figura. 4. Dimensión Interfaz.

PI5: ¿Qué tipos de investigación se encuentra en los artículos?

La mayoría de los artículos analizados corresponden al tipo "evaluación" y en segundo lugar al tipo de artículos que proponen una solución (Ver Figura 5).

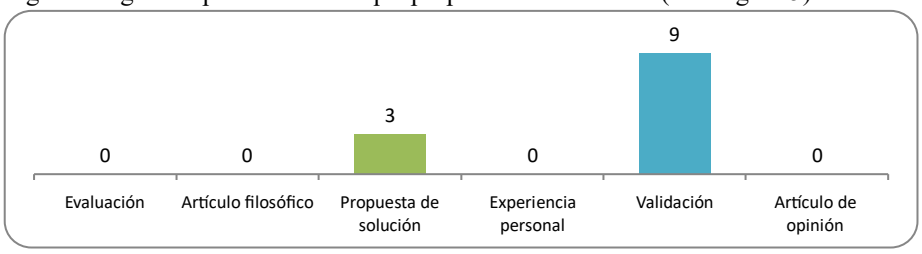


Figura. 5. Dimensión Tipo de Investigación.

4. Conclusiones y trabajos futuros

Se logró construir el estado del arte respecto al aprendizaje de la Revolución de Mayo en nivel primario mediante un Chatbot de Lenguaje Natural a través del desarrollo de un SMS. Se analizaron 12 estudios primarios y se concluye que:

- Conocimiento limitado de la temática.
- Información limitada o incompleta en los artículos sobre las herramientas.
- La poca cantidad de artículos primarios se debe a que muchos trabajos de investigación encontrados no dan respuestas a las preguntas de investigación definidas en el protocolo de revisión del SMS.
- La mayoría de los Chatbots están orientados a texto y existe una gran variedad de herramientas o tecnologías para programar los algoritmos de Lenguaje Natural.

Como futuro trabajo para continuar el desarrollo de la tesis, se construirá un Chatbot, utilizando un script en Python que mediante Machine Learning permita con una base de conocimiento responder preguntar utilizando Lenguaje Natural. El alcance del mismo tendrá como base de conocimientos la información de la Revolución de Mayo (1806 – 1810) de la República Argentina [1].

Referencias

[1] Educación Primaria. 2. Currículo de Escuela Primaria. 3. Buenos Aires. I. Siciliano, Sergio, coord. (2018)

[2] B. Kitchenham, D. Budgen y P. Brereton, Evidence-Based Software Engineering and Systematic Reviews, USA: CRC Press (2015).

[3] Wieringa, R., Maiden, N., Mead, N., Rolland, C. Requirements engineering paper classification and evaluation criteria: a proposal and a discussion. Requirements Engineering, 11(1), pp. 102-107 (2006).

[4] Orciari S., Panizzi M., Sattolo I. Apéndice. Hacia el aprendizaje de la Revolución de Mayo en nivel primario mediante un Chatbot de Lenguaje Natural. Disponible en: https://www.doi.org/10.6084/m9.figshare.23681760 (2023).

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Sistema IoT para Control y Mantenimiento de Cultivos Hidropónicos NFT

Ariana Mazzini[1], Facundo Ortega[1], Matias Solana Mendez[1], Melisa Kuzman[1,2]

[1] Universidad CAECE,
Departamento de Sistemas, Mar del Plata
[2] Universidad Nacional de Mar del Plata,
Departamento Electrónica y Computación, Mar del Plata
{ari.mazzini.98,facu98ortega,maty.mendez16}@gmail.com, melisakuzman@fi.mdp.edu.ar

Resumen. Este proyecto abarca la investigación y la creación de un sistema innovador que se encarga de supervisar y regular de manera continua las condiciones de un sistema de cultivo hidropónico, en particular, aquellos que utilizan la técnica de película nutritiva (NFT). La meta es proporcionar datos precisos sobre los factores que influyen en el ciclo de cultivo, al mismo tiempo que permitir la configuración remota de las funciones del sistema de acuerdo a las necesidades y preferencias del usuario. Para este proyecto, se implementaron tecnologías IoT, tales como diversos componentes electrónicos y sensores Arduino, y un servidor MQTT, así como también herramientas de programación tales como Spring boot para crear microservicios, y Android Studio para crear una aplicación móvil capaz de interactuar con el usuario en cuanto a visualización de datos y configuraciones referidas al sistema.

Palabras clave: IoT, Hidropónia, Sistemas NFT, MQTT, API, Aplicación móvil.

1. Introducción

1.1 Hidroponia y Sistemas NFT

La hidroponía es una forma de agricultura sin suelo que permite cultivar plantas en estructuras diseñadas para proporcionar las condiciones óptimas de crecimiento, independientemente de las dimensiones o la calidad del suelo. Entre sus beneficios se destacan el ahorro y conservación de agua, la eficiencia en el uso de recursos y la reducción del uso de pesticidas. El sistema de hidroponía más popular es el NFT (Nutrient Film Technique), que consiste en circular una fina capa de solución nutritiva a través de las raíces de las plantas en canales de cultivo, lo que optimiza el espacio y permite una producción eficiente de cultivos[1] (ver Fig. 1).

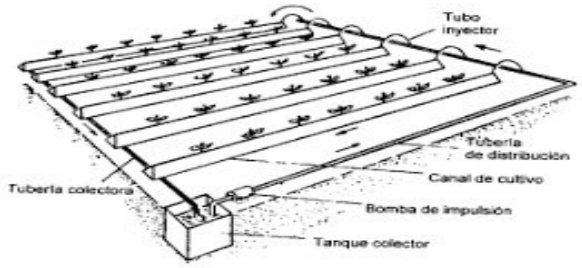


Fig. 1. Ilustración referencial a un sistema hidropónico de tipo NFT.

1.2 Proyecto

Se propone el estudio y desarrollo de un sistema capaz de realizar un control y monitoreo constante de las condiciones de un sistema de cultivo hidropónico, más específicamente de tipo NFT (Técnica de película nutritiva). El sistema brinda información fehaciente de las condiciones que afectan al circuito de cultivo y permite la configuración remota de sus funciones según las especificaciones del usuario.

Entre las características más destacables respecto del acceso remoto son: Conexión Wi-Fi, servidor web embebido, visualización de los parámetros en dispositivos móviles de diferentes características del circuito, acceso al menú de configuraciones, sincronización con reloj y calendario actual. En cuando a las funciones de monitoreo y registro del dispositivo IoT, permiten el mantenimiento de las características más relevantes influyentes al circuito: factores externos ambientales como temperatura, humedad, intensidad de la luz, y a su vez, factores del sistema, como pH y Ce de la solución nutritiva, tiempos de riego, temperatura y oxigenación de la solución, nivel del agua en el tanque y flujo del agua.

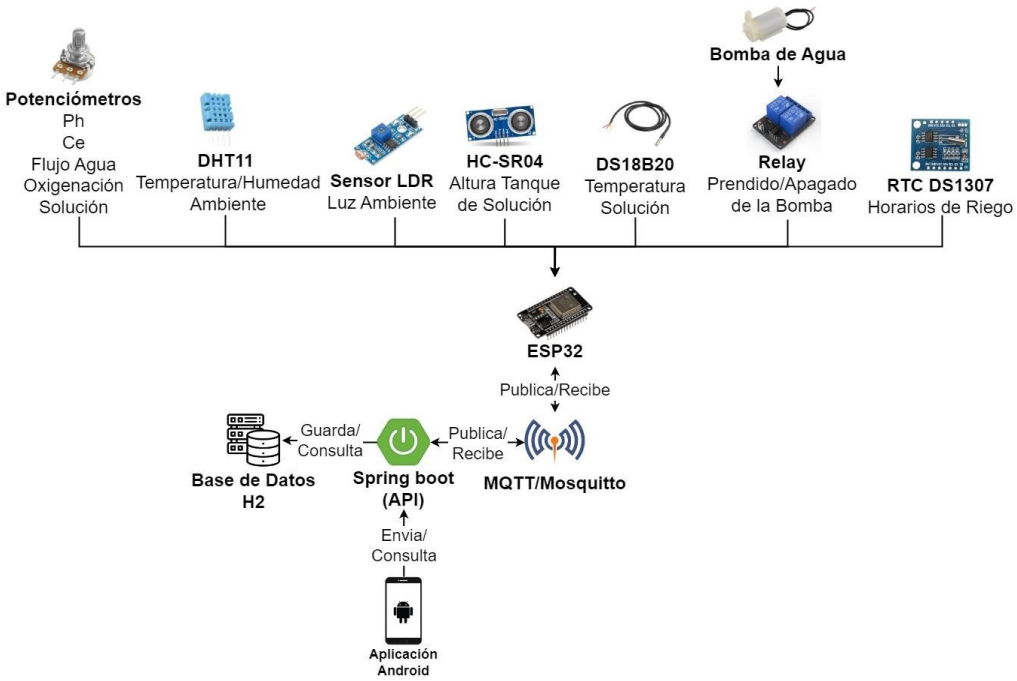


Fig. 2. Esquema general del funcionamiento del sistema.

2. Arquitectura

2.1 Hardware

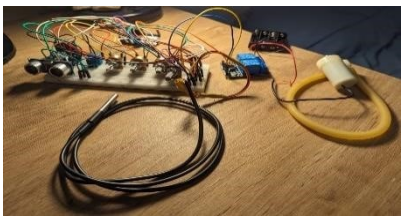


Fig. 3. Sistema físico montado en una protoboard, en conjunto a los sensores y componentes mencionados

El sistema se basa en un microcontrolador ESP32[2] que se conecta a varios sensores para medir las propiedades del ambiente y del sistema hidropónico. Este dispositivo es un circuito integrado programable con diversas entradas que permiten recibir datos digitales y analógicos de los sensores. Estos sensores captan magnitudes físicas y otras alteraciones del entorno. La Fig. 3 muestra el microcontrolador conectado a los sensores para medir las variables del ambiente y la solución nutritiva. Allí se observan todos los sensores y módulos que fueron posibles conectar al sistema, y que

se utilizan para medir correctamente las características básicas del sistema hidropónico: un sensor digital DHT11 de temperatura y humedad, un sensor de Luz Ambiente LDR, un sensor de ultrasonido HC-SR04 (para medir la cantidad de solución dentro del tanque), un sensor de temperatura sumergible DS18B20, una mini bomba de agua sumergible, junto con un relay para prenderla y apagarla, además de un módulo RTC para controlar los tiempos de riego. Por otro lado, también se utilizaron potenciómetros para simular algunos sensores más que no fue posible implementar de manera real, tales como un sensor de Ph, sensor de conductividad eléctrica (Ce), la oxigenación dentro de la solución nutritiva y el flujo del agua.

Para habilitar la comunicación entre el servidor y la placa ESP32, se implementó el protocolo **MQTT** mediante un broker de mensajería open source llamado "Mosquitto". MQTT (Message Queuing Telemetry Transport) es un estándar ISO que permite el intercambio de mensajes entre dispositivos IoT siguiendo el modelo productorconsumidor. En este modelo, los dispositivos pueden actuar como publicadores y suscriptores de mensajes. Los publicadores crean mensajes encapsulados en temas específicos, y estos mensajes solo llegan a los dispositivos suscriptores que estén interesados en esos temas. De esta manera, se establece una comunicación eficiente y selectiva entre los dispositivos conectados.

2.2 API

El sistema desarrollado utiliza un servidor Back End programado en Java 11 y el framework Spring Boot[3] para crear una API REST[4]. Esta API se conecta al servidor MQTT y recibe información del microcontrolador, el cual publica sus datos en el servidor. La API almacena temporalmente los mensajes recibidos en una variable global y luego los guarda en una base de datos SQL en memoria RAM llamada H2. Esta frecuencia de almacenamiento se optimiza para mejorar los recursos y los tiempos del servidor. La base de datos H2 almacena los registros con fecha y hora de las mediciones, relacionándolos a través de usuarios y dispositivos para permitir la gestión personalizada de cada dispositivo preferido por el usuario. Además de esto, dentro de la base de datos se almacena toda la información necesaria para el correcto funcionamiento del sistema, y que será utilizada a través de todo el ciclo de la aplicación y del sistema conjunto, como por ejemplo, los datos del usuario para poder acceder a la aplicación, los datos tomados en tiempo real desde el microcontrolador y sensores, las configuraciones guardadas por cada usuario, entre otras cosas.

Además, el servidor implementa seguridad mediante un token basado en el protocolo JWT[5], un estándar abierto que garantiza la transmisión segura de información entre las partes involucradas. Este token se utiliza en todas las peticiones después de que el usuario inicia sesión para verificar su identidad y evitar accesos no autorizados.

La aplicación móvil, a un nivel más alto, consume los endpoints creados en la API utilizando el protocolo REST. Los endpoints son URLs con condiciones específicas para obtener y presentar información en diversos formatos, como JSON, en las vistas a las que accede el usuario. En este caso, se utilizan endpoints representativos para cada acción que se desee realizar desde la aplicación móvil, siendo algunos ejemplos:

- POST /user/login: Permite a un usuario registrado iniciar sesión enviando su

correo electrónico y contraseña. La respuesta incluye los datos del usuario y un token de autenticación.

- POST /user/register: Permite registrar un nuevo usuario proporcionando su correo electrónico, contraseña y nombre de usuario. La respuesta es un código HTTP 204 (No Content) que indica que la creación se ha realizado con éxito.
- GET /info/realtime: Permite a un usuario logueado recibir la información más reciente de las mediciones de los sensores enviando su token. La respuesta incluye las últimas mediciones de cada sensor junto con la fecha y hora de la última actualización.
- PUT /threshold/limits: Permite a un usuario logueado cambiar los valores de los sensores enviando una lista con los nuevos valores para cada sensor. La respuesta es un código HTTP 200 OK.
- GET /threshold/default-limits: Permite a un usuario logueado ver los valores por defecto de todos los sensores enviando su token. La respuesta es una lista de los valores por defecto de los sensores.

2.3 Aplicación

La aplicación móvil permite visualizar la información recolectada desde el microcontrolador en tiempo real, y a su vez, configurar características personalizadas, como límites de alerta y tiempos de riego diarios. Todas las pantallas están realizadas a través del entorno de desarrollo Android Studio [6], el cual permite no solo configurar las pantallas, sino también editar sus diseños y crear un emulador de cualquier dispositivo con sistema operativo Android para ejecutar la aplicación.

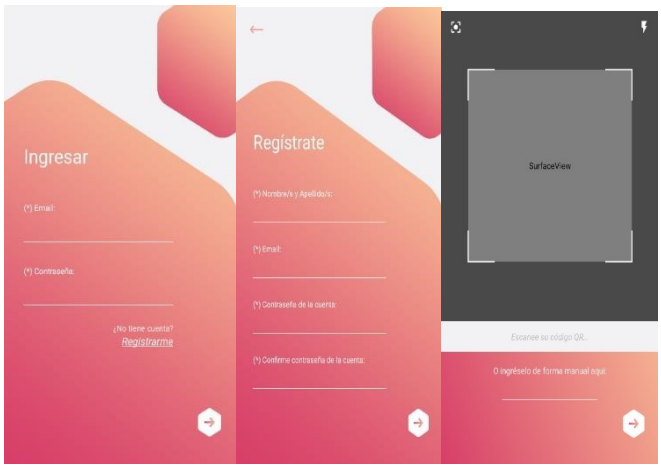


Fig. 4. De izquierda a derecha: a. Pantalla de Bienvenida, b. Pantalla de Registro y c. Pantalla del Scanner QR.

Para ingresar a la aplicación, se debe contar con una cuenta de usuario. Para crearla, se registrar con datos personales (email, nombre, apellido, contraseña) y,

opcionalmente, se permite registrar y/o adjuntar un código de dispositivo, el cual es representativo para el microcontrolador que nos permite guardar y visualizar toda la información del sistema y del ambiente (ver Fig. 4). Una vez se crea el usuario, basta con iniciar sesión y la aplicación redirige hacia el menú principal, el cual cuenta con un panel que permite visualizar todos los componentes conectados al microcontrolador. En la aplicación se pueden observar los diferentes parámetros internos y externos al sistema, así como sus estados. Los estados son lo suficientemente representativos; si el componente está dentro de los límites deseados, se mostrará en verde, si esta sobre los límites deseados se mostrará de color amarillo, y de lo contrario, si se excede sobre lo permitido, se marcará de color rojo. A su vez, se enviará una notificación al dispositivo móvil en caso de que alguno de los datos registrados por el dispositivo físico de monitoreo se exceda de los umbrales o límites preestablecidos (ver Fig. 5). Debido a que el sistema puede ser utilizado para diferentes cultivos, el panel de configuración permite al usuario modificar tiempos de riego, umbrales de alertas, tipo de notificaciones, entre las características más relevantes.

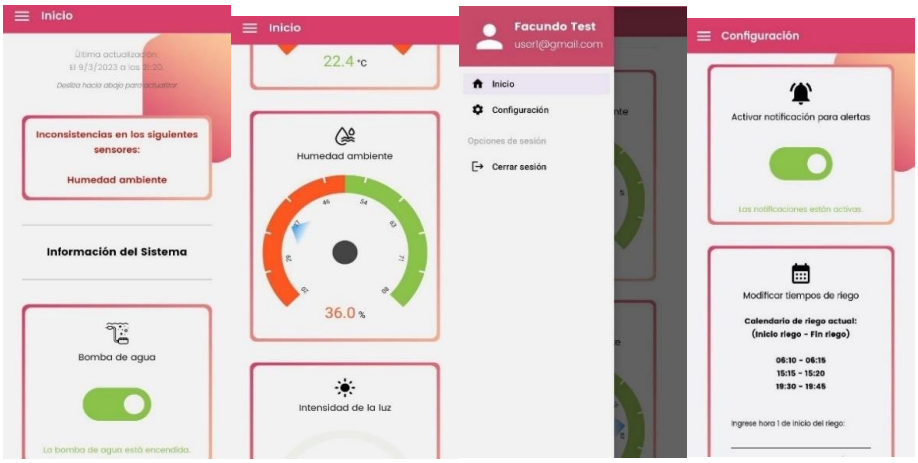


Fig. 5. De izquierda a derecha: a. Estados de alerta, b. Variable en alerta, c. Menú desplegable y d. Pantalla de Configuración.

3. Conclusión

El objetivo principal de este proyecto es proporcionar información en tiempo real de los factores clave para cultivos hidropónicos. Las ventajas de su uso son significativas para los emprendedores de los cultivos hidropónicos, brindando mayor accesibilidad y comodidad al permitir el monitoreo remoto. Su flexibilidad para configurar umbrales de alertas para prevenir condiciones adversas permite que este sistema pueda ser utilizado en diversos cultivos y condiciones climáticas. Este sistema presenta ventajas competitivas significativas, ya que brinda una solución integral, de bajo costo y fácil de usar, que puede atraer a un amplio rango de cultivadores, desde pequeños emprendedores hasta operaciones comerciales a gran escala.

Por otro lado, el proyecto presenta diversos puntos de mejora que podrían considerarse para una mejora en la implementación del sistema. Entre ellos se

encuentran la capacidad de controlar la composición de la solución nutritiva, incluir análisis estadísticos de los datos históricos adquiridos, la integración de varios dispositivos físicos al mismo espacio, además de poder asociar varios dispositivos a cada usuario, lo que permitiría supervisar y administrar diferentes configuraciones de cultivos en una sola plataforma web.

Referencias

1. Una Guía para Principiantes en Hidroponía - hydrocultura, https://hydrocultura.com/blogs/news/una-guia-para-principiantes-en-hidroponia.
2. Microcontrolador Esp32, naylampmechatronics.com/espressif-esp/384-nodemcu-3230-pin-esp32-wifi.html
3. Baldung - Guia de Spring boot, https://www.baeldung.com
4. ¿Qué es una API de REST? - Red Hat, https://www.redhat.com/es/topics/api/what-isa-rest-api
5. JWT - JSON Web Token, https://jwt.io/introduction
6. Documentation - Android Developers, https://developer.android.com/docs?hl=es-419

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

BreatheEasy: Sistema de monitoreo ambiental distribuido para la detección y alerta temprana de fugas de gases tóxicos

Franco Bandieri[*], Nicolás Falcone[*], Melisa Kuzman[*,†]

[*] Dpto. de Sistemas, Universidad CAECE, Mar del Plata

[†] Dpto. de Electrónica y Computación, Universidad Nacional de Mar del Plata, Mar del Plata
{nicolas.a.falcone99,francobandieri205}@gmail.com melisakuzman@fi.mdp.edu.ar

Resumen. Este trabajo propone el desarrollo de un sistema de monitoreo para la detección y alerta temprana de fugas de gases tóxicos como monóxido de carbono, metano, propano y butano, basado en tecnologías IoT. Para ello se utilizan dispositivos inalámbricos conectados a una red WiFi que permiten una distribución y localización más conveniente para gran variedad de aplicaciones. Se presentan las bases para el desarrollo de una interfaz gráfica moderna y amigable para el usuario final. Además, gracias al uso de tecnología presente en el mercado, el sistema brinda la posibilidad de agregar fácilmente sensores predeterminados para medir variables que pueden ser de interés para algunas aplicaciones, tales como humedad, temperatura, detección de movimiento, entre las más relevantes.

Keywords: IoT, MQTT, Microcontrolador, Gases, CO, Seguridad.

1. Introducción

Siempre que está presente la combustión de gases como fuente de energía, se pueden generar elementos altamente tóxicos o peligrosos para la vida humana y puede darse tanto en hogares, fábricas, negocios pequeños, entre otros. Algunos de esos gases son el monóxido de carbono (CO), gases naturales (propano y butano) y metano. Al llegar el invierno aumentan los casos de muertes por intoxicación con monóxido de carbono producto del uso de aparatos viejos o defectuosos que implican un riesgo certero a la vida cuando se encienden, tal es así que constituye una de las principales causas de muerte por envenenamiento para todas las edades. El Ministerio de Salud de la República Argentina estima unas 250 muertes por año debidas al CO; solo en 2018 se registraron 2.600 casos de este tipo de accidentes [1].

Hoy en día se cuenta con tecnologías que nos permiten enfrentar esta problemática de distintas formas. En este trabajo se muestra el desarrollo de un sistema distribuido con el propósito de monitorear, alertar y prevenir potenciales fugas de gases tóxicos, haciendo uso de tecnologías como IoT. El sistema está compuesto por sensores, encargados de medir los niveles de los gases, la temperatura y humedad en el ambiente, que se encuentran conectados al microcontrolador ESP32[2], y cuyos datos son enviados a un *broker* MQTT[3], llamado HiveMQ[4], para luego ser visualizados en la aplicación WEB alojada en la nube (ver Fig. 1).

2. Arquitectura del sistema

2.1 Plataforma IoT

El firmware del proyecto se desarrolló en C++ y se instaló en un µCU ESP-WROOM-32, del cual se utiliza su memoria Flash no solo para almacenar el código del programa, sino también todos los datos de su configuración, como el tiempo entre muestras, el estado del dispositivo y un identificador unívoco, lo que permite que no se pierdan datos ante una falla en la alimentación del dispositivo. Se elige este dispositivo como plataforma debido a su bajo costo, bajo consumo y módulo Wi-Fi integrado.

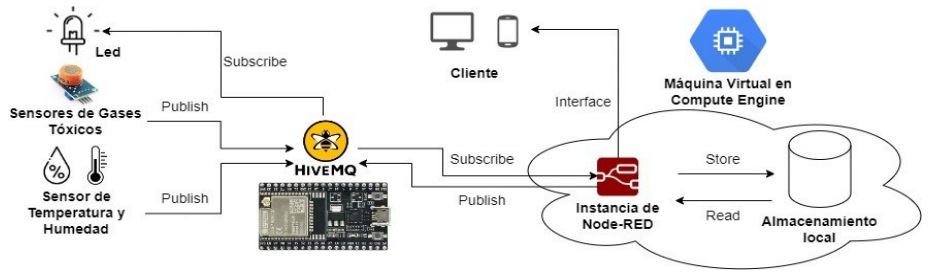


Fig. 1. Esquema del sistema.

Por otro lado, los usuarios tienen la opción de elegir qué sensores estarán activos en la plataforma, los cuales pueden ser: MQ-4 Gas Natural Metano [5], MQ-6 Gas Propano Butano [6], MQ-9 Gas Combustible y Monóxido de Carbono [7]; y DHT22 de Temperatura y Humedad [8]. A su vez, cada plataforma cuenta con una salida digital a un LED para indicar su estado (encendido / apagado).

2.2 Aplicación

Para desarrollar, integrar y gestionar la aplicación junto con el resto de sus componentes, se utiliza una herramienta de programación basada en nodos, llamada Node-RED [9], que permite crear flujos de trabajo entre diferentes tecnologías y aplicaciones basadas en eventos e IoT de manera eficiente y veloz. Es allí donde está alojada la estructura y este proyecto en sí mismo.

En cuanto a la conexión entre los clientes y el servidor, se implementa el protocolo MQTT, el cual se ha convertido en uno de los pilares del IoT por su sencillez y ligereza. Para filtrar los mensajes que son enviados a cada cliente, los mismos se disponen en *topics* organizados jerárquicamente. El *broker* utilizado es HiveMQ, y consiste en una plataforma de mensajería basada en clientes diseñada para un flujo eficiente, confiable y rápido de los datos desde y hasta diversos dispositivos IoT. Es así que, en base al sensor y a un identificador único definido para cada placa (compuesto por la dirección MAC de cada dispositivo, el domicilio del establecimiento donde se instalará este producto, así como también una ubicación más detallada del mismo, como un piso, departamento o sector), se construye el *topic* al cual son enviados los datos recolectados, de modo que los mismos son transportados de manera independiente, evitando así problemas de interferencia y/o sobreescritura (ver Fig. 2).

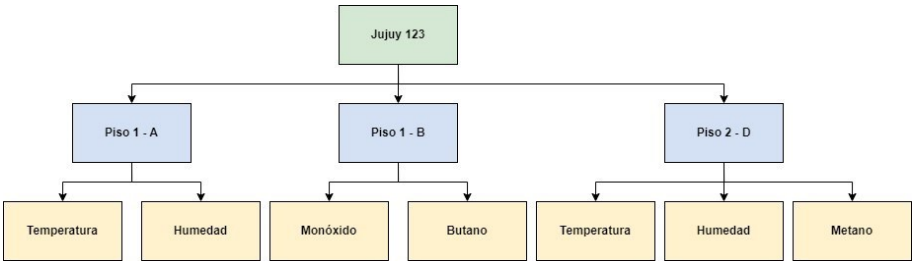


Fig. 2. Jerarquía de *topics*.

Por otro lado, en cuanto al alojamiento del servidor de Node-RED, se optó por la implementación de un servicio disponible dentro de *Google Cloud Platform* (GCP)[10], llamado *Compute Engine*. Es a través de este que se inicializa una máquina virtual cuyo fin es mantener en constante ejecución la instancia del servidor de Node-RED, de modo que pueda ser accedido desde cualquier ubicación y dispositivo. La disponibilidad y la seguridad del sistema están garantizadas por el servicio provisto por Google ya que ofrece características de seguridad robustas, escalabilidad, alta disponibilidad, opciones de copias de seguridad y recuperación ante fallas; además de soporte técnico confiable; convirtiéndolo así en una opción sólida y segura para el hosting de aplicaciones y servicios en la nube.

Del mismo modo, en relación a los datos generados por el sistema, se optó por el almacenamiento local de los mismos en el servidor. Dichos datos, serán almacenados en archivos de formato JSON (*JavaScript Object Notation*) debido a su facilidad de implementación, organización y facilidad de lectura. Además, en dicha instancia serán almacenados todos los usuarios registrados en el sistema en un archivo encriptado utilizando el algoritmo TripleDES y añadiendo así una capa extra de seguridad a los mismos, sumada a la robustez brindada por Google en todos sus servicios.

Finalmente, para el desarrollo del *frontend* del sistema, se implementó una librería dentro de Node-RED, llamada UIBuilder[11], que establece una interfaz entre el *backend* y código Javascript en conjunto con el *framework* Bootstrap-Vue[12]; de manera que se hace posible desarrollar una aplicación basada en la WEB.

2.3 Base de datos

La estructura de carpetas que funciona como base de datos local, almacenada en la nube junto con el servidor de Node-RED, puede representarse como se muestra en la Fig. 3:

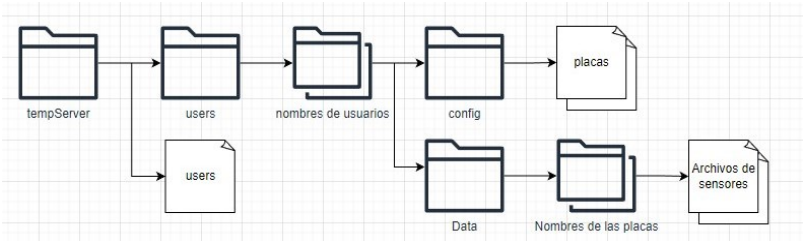


Fig. 3. Estructura de carpetas (base de datos local)

Una carpeta principal llamada tempServer donde puede encontrarse un archivo .JSON encriptado con los datos de todos los usuarios registrados y una carpeta llamada *users* dentro de la cual se crea otra por cada usuario, identificadas con el nombre de los mismos. A su vez, dentro de cada carpeta de usuario se repite la estructura de un directorio llamado *config* donde se pueden encontrar múltiples archivos .JSON cuya identificación es determinada por el nombre de cada placa que el usuario tenga asignada. Estos archivos contienen toda la configuración de una placa en particular (ID, tiempo entre muestras, sensores conectados y sus valores de alerta, entre otros).

Por otro lado, se encuentra la carpeta *Data*, dentro de la cual son almacenados múltiples directorios, también representados por los nombres de las placas del usuario. Dentro de cada uno, se encuentran almacenados archivos de extensión .CSV (uno por cada sensor asignado a una placa específica) donde se registran los valores leídos por la placa. De esta forma, y en cada adquisición de datos que realiza la plataforma, en el archivo .CSV se creará una nueva línea con la fecha de registro y el valor detectado.

2.4 Interacción con el usuario

Cuando el usuario ingresa al sistema a través de la instancia alojada en la nube, se encuentra con una pantalla de inicio de sesión. Con la sesión activa, pasa a la pantalla donde puede visualizar los dispositivos y sectores que tiene asignados, los estados de cada uno y las alertas activas (ver Fig. 4).

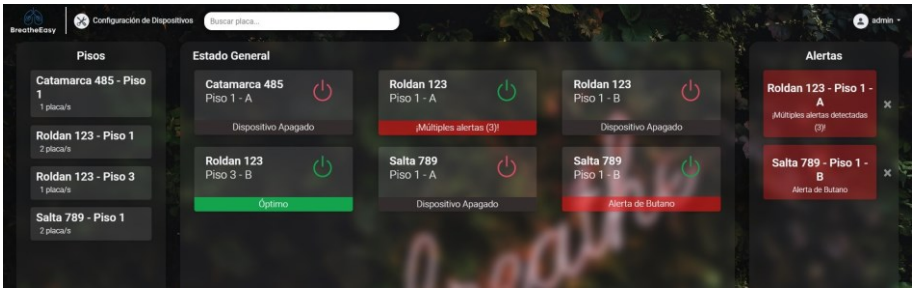


Fig. 4. Pantalla Principal

Por otra parte, el usuario puede acceder a información más detallada de un dispositivo donde se ven los últimos datos leídos por el mismo y es posible descargar un reporte de todos los datos en un periodo seleccionado; o de un sector, donde puede ver todos los dispositivos presentes en un sector, sus estados de alerta e información promedio de estos (ver Fig. 5). Es importante destacar que, en el caso de que se presente un evento no deseado, el usuario es notificado a través de un e-mail enviado a la dirección de correo asociada al dispositivo que presenta la alerta.



Fig. 5. Pantallas de Información

3. Conclusiones

El sistema BreatheEasy desarrollado en el presente artículo busca reducir el peligro de las fugas de gases tóxicos y los riesgos que los mismos pueden presentar frente a la exposición humana. Este sistema brinda seguridad, tranquilidad y control a sus usuarios a través de una interfaz sencilla e informativa, donde cada uno tendrá la posibilidad de acceder rápida y eficazmente a toda la información que pueda necesitar junto con la opción de adaptar cada dispositivo para cumplir con sus expectativas. Además, se destaca que para su implementación fueron elegidos sensores y plataformas de gran accesibilidad y bajo costo en el mercado local.

Actualmente, este sistema cumple el rol de detección y alerta temprana ante la fuga de gases, pero tiene el potencial para escalar sus capacidades de modo que pueda actuar ante el peligro detectado de manera automatizada, ya sea mediante la apertura de persianas, el encendido de sistemas de ventilación, el cierre de la conexión de gases en todo el recinto e incluso comunicación directa con los servicios de emergencia designados.

Como trabajo a futuro, se considera la implementación de una base de datos no relacional, por ejemplo *Google Cloud Firestore*, logrando así una mayor capacidad de escalabilidad y velocidad de respuesta a las peticiones realizadas del lado de los clientes; así como también la implementación de un paso extra de autenticación sobre los clientes que envían mensajes MQTT de modo que, previo al envío de los mismos, deben establecer una conexión con el *broker*, para que evalúe sus credenciales y acepte o no esa solicitud. En este proyecto no se implementa dicha funcionalidad ya que no es posible hacer uso de la misma en su versión gratuita sin instalar un paquete local de

HiveMQ, sin embargo, debe tomarse en cuenta para reforzar la seguridad en las comunicaciones entre la aplicación y los clientes, y así aumentar considerablemente la fiabilidad del sistema.

Referencias

1. Honorable Cámara de Diputados de la Nación, "PROYECTO DE DECLARACIÓN Expediente 3265-D-2019", https://www.diputados.gob.ar/proyectos/proyecto.jsp?exp=3265-D-2019
2. Espressif Systems, "Hoja de datos ESP32", https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf
3. MQTT.org, "MQTT: The Standard for IoT Messaging", https://mqtt.org/
4. HiveMQ GmbH, "HiveMQ - Enterprise ready MQTT to move your IoT data", https://www.hivemq.com/
5. Hanwei Electronics, "Hoja de datos MQ-4", https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-4.pdf
6. Hanwei Electronics, "Hoja de datos MQ-6", https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-6.pdf
7. Hanwei Electronics, "Hoja de datos MQ-9", https://www.electronicoscaldas.com/datasheet/MQ-9_Hanwei.pdf
8. "Hoja de datos DHT22", https://www.mouser.com/datasheet/2/737/dht-932870.pdf
9. OpenJS Foundation, "Node-RED", https://nodered.org/
10. Google, "Cloud Computing Services | Google Cloud", https://cloud.google.com/
11. OpenJS Foundation, "Node-RED UIBuilder", https://flows.nodered.org/node/node-red-contrib-uibuilder
12. BootstrapVue, https://bootstrap-vue.org/

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Tecnología colaborativa para investigadores: implementación y optimización de un repositorio GitLab para gestión de la configuración y versionado

Damián Ferrara[1], Emanuel Rodriguez[2], Romeo Lorenzo Monfroglio[3], Manuel Robles[4], María Ayelén Díaz Lapérgola[5], Leopoldo Nahuel[6]

GIDAS - Grupo de I&D Aplicado a Sistemas informáticos y computacionales
Universidad Tecnológica Nacional (UTN) - Facultad Regional de La Plata (FRLP)
Av. 60 esq. 124 s / nº CP 1900, La Plata, Buenos Aires, Argentina
[1]dferrara@frlp.utn.edu.ar, [2]errodriguez@frlp.utn.edu.ar, [3]rmonfrolio@alu.frlp.utn.edu.ar, [4]mroblesrobles@alu.frlp.utn.edu.ar, [5]madlapergola@frlp.utn.edu.ar, [6]lnahuel@frlp.utn.edu.ar

Resumen: Partiendo de la instalación del servidor Lenovo ThinkServer SR530 en el laboratorio, Grupo de Investigación y Desarrollo Aplicado a Sistemas Informáticos y Computacionales (GIDAS) a cargo del grupo "Transformación digital en organizaciones aplicando tecnologías de infraestructura informática basada en computación en la nube" (TDO), se presentaron nuevos desafíos relacionados con la descentralización y versionado de información y código.

Para dar solución a estos, el grupo anteriormente mencionado ha investigado sobre herramientas de gestión de la configuración y decidió implementar GitLab por su licencia de software de código abierto, amplia popularidad, uso y documentación existente acerca de ella.

La implementación de GitLab en nuestra infraestructura logrará una significativa mejora en la gestión y control de la información, al centralizar los repositorios en nuestro servidor; aseguramos un acceso más eficiente a los proyectos y documentos importantes, evitando la dispersión y pérdida de información que podría ocurrir. Además, podremos facilitar la colaboración entre los equipos y garantizar la seguridad y confidencialidad.

La adopción de esta herramienta fortalecerá el control, coordinación y eficiencia del grupo. A pesar de ciertas dudas iniciales, este último se adaptó y niveló conocimientos para aprovechar los beneficios que puede proporcionar GitLab en el servidor para todo el laboratorio.

Palabras Claves: gestión de la configuración, GitLab, repositorio, versionado.

1. Introducción

En búsqueda de eficiencia, agilidad, control y gestión en el ámbito de investigaciones científica del Grupo de Investigación y Desarrollo Aplicado a Sistemas Informáticos y Computacionales (GIDAS) ha llevado al laboratorio a adquirir el servidor Lenovo ThinkServer SR530 [1], un hito significativo en el crecimiento del mismo, que ha permitido al grupo disponer de una infraestructura propia. El equipo "Transformación digital en organizaciones aplicando tecnologías de infraestructura informática basada en computación en la nube" (TDO) comenzará a prestar servicios entorno a infraestructura IT.

En el artículo “Una experiencia de implementación de infraestructura informática: recorriendo el camino desde lo académico hasta la instalación y puesta en funcionamiento” [2], publicado en el año 2022, se describe el fin con el que fue instalado este equipo.

Sin embargo, tras esta implementación, el grupo se enfrenta a desafíos importantes relacionados con la descentralización y el versionado, tanto en la gestión de información como en el desarrollo de código fuente. Para dar solución a esta problemática realizamos una investigación de herramientas que nos permitan solucionarlo de manera sencilla y eficaz.

Por otro lado, también buscamos una herramienta que nos brinde el servicio de “Tickets”, lo que nos facilitará la administración y seguimiento de problemas que puedan surgir. Además, la capacidad de versionado de archivos que nos posibilite la modificación de estos, permitiendo tener un historial completo de cambios y la posibilidad de retroceder a versiones anteriores si es necesario. Esta característica es invaluable para mantener un registro preciso de los progresos, solucionar problemas y evitar confusiones en el trabajo colaborativo.

Adicionalmente, poder otorgar diferentes niveles de acceso y permisos, para asegurarnos de que cada equipo acceda únicamente a la información relevante para sus tareas, garantizando la seguridad y confidencialidad de los datos sensibles.

Asimismo, el equipo TDO podrá prestar estos servicios personalizados de configuración para el laboratorio. Lo que implica que los diferentes equipos del GIDAS podrán tener entornos de trabajo adaptados a sus necesidades específicas, lo que mejorará la eficiencia y la productividad.

2. Aportes realizados

Contar con una herramienta de gestión para la configuración proporciona varios beneficios a nuestro laboratorio de investigación.

La problemática principal es la descentralización de la configuración, ya que cada equipo de investigación utiliza su propio repositorio en distintos proveedores de la nube, como GitHub y GitLab. Estos son creados por becarios o el coordinador del proyecto, lo que puede ocasionar conflictos de acceso o dificultades para la gestión de los usuarios cuando el administrador original deja su función.

Al tener instalada una herramienta de gestión de la configuración en nuestro servidor local, obtendremos un mayor control y coordinación con los distintos equipos de investigación. Podremos identificar los elementos de configuración que cada equipo necesita y ofrecer un servicio más personalizado. Además, brindaremos capacitación, recomendaciones y mejores prácticas para el uso de la gestión de la configuración.

Esta herramienta nos permitirá tener un control exhaustivo de las versiones y cambios de la configuración, con soporte dedicado para este servicio. Además, el equipo de investigación TDO, lleva más de cuatro años capacitándose en infraestructura, lo que nos permitirá ofrecer un servicio y soporte profesional, algo que no siempre se encuentra en los proveedores gratuitos que se están utilizando.

Asimismo, podremos tener un control más sólido en la gestión de las baselines, siguiendo las mejores prácticas y estándares como normas ISO e ITIL, entre otros.

Con esta implementación también podremos comenzar a automatizar gradualmente nuestras tareas de gestión de la configuración. Esto implica realizar diversas acciones de forma automática mediante el uso de herramientas específicas, como sistemas de control de versiones, sistemas de seguimiento de problemas y plataformas de integración continua con GitLab CI.

3. Implementar

Para dar solución a los desafíos presentados en el grupo GIDAS se ha investigado dentro del equipo TDO sobre herramientas de gestión de la configuración; se decidió implementar GitLab [3] por su amplia popularidad, uso y documentación existente acerca de ella. GitLab es un manejador de repositorios Git basado en web con manejo de wiki y seguimiento de tickets.

Con el fin de mantener un historial de versiones, en todo momento se implementó esta herramienta para versionar el código y la documentación necesaria en los distintos equipos de investigación.

Los repositorios en ella son espacios donde se almacena el código fuente de un proyecto. Estos funcionan utilizando el sistema de control de versiones Git, que registra y administra los cambios en el código a lo largo del tiempo. Los desarrolladores pueden clonarlo en sus máquinas locales, realizar cambios en el código y luego enviar esos cambios de vuelta al repositorio.

Estos últimos permiten organizarse en ramas, conocidos también como “branch”, que son copias independientes del código. De este modo los desarrolladores pueden trabajar en paralelo en diferentes características o soluciones sin afectar la rama principal.

Al poseer un control de versiones contamos con el beneficio de un análisis y seguimiento de los cambios realizados, podemos comparar el funcionamiento con versiones antiguas y tenemos la capacidad de identificar los inconvenientes para su futura solución.

Si hablamos de colaboración este se logra mediante comentarios, revisiones de código y aprobaciones. Varios usuarios pueden trabajar en el mismo repositorio, realizar cambios en diferentes branch y fusionar sus cambios cuando sea necesario. GitLab también proporciona opciones de gestión de permisos para controlar el acceso y las acciones permitidas en los repositorios.

Además, incluye funciones de integración continua (CI) que permiten automatizar tareas como compilar, probar y desplegar el código en diferentes entornos. Los repositorios se pueden configurar para ejecutar pipelines de CI/CD [4] automáticamente cuando se realizan cambios en el código.

En GitLab tenemos el sistema de issues que son un punto donde se inicia una conversación sobre el código, permiten discutir la implementación de una nueva idea, sugerir propuestas de features, reportar bugs y fallos, obtener reportes y planear implementaciones de código [5].

Es una herramienta que nos deja gestionar y dar seguimiento a problemas, tareas y solicitudes relacionadas con proyectos de desarrollo de software. Los usuarios pueden crear tickets para informar sobre problemas o solicitar new features, asignar responsables, establecer prioridades y agregar comentarios. Estos se organizan en un tablero de proyecto, donde se pueden visualizar y gestionar de forma eficiente. Los miembros del equipo pueden colaborar en estos mediante comentarios, debates y adjuntando archivos. A medida que los tickets avanzan, se actualizan para reflejar el progreso, como cambiar el estado, registrar tiempo dedicado y vincular con "commits" o solicitudes de extracción. GitLab también se integra con otras herramientas y permite automatizar acciones y notificaciones.

4. Proyecto, relevamiento y desarrollo de experiencia

Cuando emprendimos esta experiencia y decidimos incursionar en GitLab para el servidor del GIDAS se nos presentaron diversas dudas a priori: ¿qué tipo de información necesitan los equipos que sea gestionada? ¿Cómo los demás grupos del GIDAS almacenan su código actualmente? ¿Cómo se usaría desde el lado administrador GitLab? ¿De qué modo se debería gestionar los usuarios y los roles? ¿Cómo administrar los tickets?

Poco a poco fuimos logrando una intensa tarea de nivelación de conocimientos y adaptación a la nueva herramienta. Al ser un grupo de personas con diferentes experiencias en IT, debimos recaudar la información necesaria para poder entender, trabajar y, posteriormente, aplicar lo aprendido a el servidor en cuestión.

La información de los integrantes actuales de nuestro laboratorio puede ser pedida mediante mail y encuestas con preguntas abiertas, de la manera: ¿Dónde guardan actualmente el versionado de su código? ¿Un administrador o varios? ¿Qué pasos siguen en caso de querer compartir entre sus miembros el código en cuestión? ¿Cuál es su plataforma de preferencia? ¿Qué opinan de un lugar centralizado de almacenamiento común?

Con esta información podemos brindar un mejor servicio para todos los equipos; una mejor organización y un flujo continuo de código firmemente almacenado en infraestructura local, donde pueda ser accesible desde cualquier lugar del que tenga acceso un grupo de investigadores.

5. Trabajo futuro

Al utilizar GitLab como herramienta en nuestro servidor podemos dar servicios de CI/CD [4]. Tendremos a disposición CI porque validamos si el código funciona correctamente antes de enviarlo a producción y así notificar a los desarrolladores de lo ocurrido. Por otro lado, aplicamos CD que consiste en que el código válido, pase a desplegarse automáticamente en un servidor en producción.

Teniendo en cuenta una filosofía basada en DevOps, tanto el equipo de operaciones como el equipo de desarrollo trabajarán en varias fases para fomentar la colaboración entre los equipos de infraestructura y desarrollo.

6. Conclusiones

Al adoptar una herramienta de gestión de la configuración en nuestro servidor local, fortaleceremos el control, la coordinación y la eficiencia en la gestión de los distintos equipos de investigación. Además, mejoraremos la calidad de nuestros servicios y seguiremos avanzando hacia la automatización para optimizar nuestras prácticas y procesos de desarrollo.

Por otra parte, se pretende lograr que la implementación sistemática de distintas herramientas dé soluciones a los diferentes equipos de investigación; teniendo como último objetivo implementar la filosofía DevOps [6] en el área de infraestructura.

Para esto, se pretende de manera escalonada ir incorporando tecnologías de contenedores, orquestadores, infraestructura como código, CI/ CD [4], entre otras.

Referencias

1. ThinkSystem SR530 Setup Guide. (2020). Lenovofiles.com. https://thinksystem.lenovofiles.com/help/topic/7X07/setup_guide.pdf
2. Ferrara, D. Nahuel, L. Di Santo, L. Basalo, A. Bertuzzi Gaspari, A. Rodríguez, E.: Una experiencia de implementación de infraestructura informática: recorriendo el camino desde lo académico hasta la instalación y puesta en funcionamiento: XXVIII Congreso Argentino de Ciencias de la Computación (CACIC), pp. 510—514. (2023) http://sedici.unlp.edu.ar/handle/10915/149422
3. GitLab Docs, https://docs.gitlab.com/ee/install/index.html
4. GitLab CI/CD, https://docs.gitlab.com/ee/ci/
5. Gitlab – Platzi, https://platzi.com/clases/1526-gitlab/19282-planificacion-en-gitlab-issues/
6. Belalcázar, A: Arquitectura de un Data Center con Herramientas DevOps, pp. 21—25. (2017) http://sedici.unlp.edu.ar/handle/10915/63838

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Propuesta de Evaluación de Experiencia de Usuario Usando Detección Facial de Emociones

Sergio E. Ledesma Vila[1] [0000-0003-1929-5315], Laura N. Aballay[1] [0000-0001-5462-7683], Flavia Millan[2] [0009-0004-7457-3837]

[1] Instituto de Informática, Facultad de Ciencias Exactas, Físicas y Naturales, Universidad Nacional de San Juan, Argentina

[2] Departamento de Informática, Facultad de Ciencias Exactas, Físicas y Naturales, Universidad Nacional de San Juan, Argentina

sergioledesma.9427@gmail.com, lnaballay@gmail.com, flavia.millan@gmail.com

Abstract. Los métodos de evaluación de Experiencia de Usuario (UX) se centran en determinar cómo se sienten los usuarios respecto al sistema de software evaluado. Existen diversos métodos de análisis a la hora de evaluar características de calidad fundamentales de UX en sistemas web. Sin embargo, las evaluaciones con los usuarios pueden verse afectadas por la forma en que se recopilan los datos. Por lo tanto, el uso de las expresiones faciales como herramienta para evaluar la dimensión emocional de manera automática, establece una herramienta no invasiva para el usuario. El presente trabajo corresponde a un trabajo final de la carrera Lic. en Computación, de la Facultad de Ciencias Exactas Físicas y Naturales de la Universidad Nacional de San Juan. Se propone, mediante detección facial de emociones, evaluar la Experiencia de Usuario mientras se utiliza un sistema.

Keywords: Experiencia de Usuario, Evaluación de Experiencia de Usuario, Detección Facial de Emociones.

1. Introducción

Los métodos de evaluación de Experiencia de Usuario (UX) se centran en determinar cómo se sienten los usuarios respecto al sistema de software evaluado [1]. Existen diversos métodos de análisis a la hora de evaluar características de calidad fundamentales de UX en sistemas web. Sin embargo, las evaluaciones con los usuarios pueden verse afectadas por la forma en que se recopilan los datos [2]. Por lo tanto, el uso de las expresiones faciales como herramienta para evaluar la dimensión emocional de manera automática, establece una herramienta no invasiva para el usuario. La presente propuesta pretende mediante detección facial de emociones evaluar la Experiencia de Usuario mientras se utiliza un sistema.

2. Justificación

Los métodos de evaluación de UX se centran en determinar cómo se sienten los usuarios respecto al sistema de software evaluado [1]. Tener este conocimiento ayudará a lograr una experiencia de usuario positiva y productos deseables.

Se necesitan métodos que midan de manera precisa la experiencia de los usuarios, que tomen en cuenta características subjetivas de la calidad y satisfacción de usuario de un producto software. UX permite determinar si un producto está siendo una experiencia placentera para el usuario [3], por esta razón, la medición y recolección de datos mediante la observación e indagación de las emociones humanas es la más pertinente.

Sin embargo, las evaluaciones con los usuarios pueden verse afectadas por la forma en que se recopilan los datos [2]. La presencia del moderador humano en una entrevista puede tener un impacto negativo en la sesión de evaluación, haciendo que los usuarios se sientan incómodos al preguntarles sobre su experiencia en persona [4].

Las tecnologías de evaluación objetiva, que recopilan información sobre la respuesta emocional de los usuarios o sacan conclusiones basadas en el comportamiento de los usuarios, pueden mitigar este problema. En este sentido, la detección facial de emociones, constituye además una herramienta transcultural [5][6].

Por lo tanto, el uso de las expresiones faciales como herramienta para evaluar la dimensión emocional en forma automática, establece una herramienta fiable, segura, no invasiva para el usuario, económica y además se puede realizar sin restricciones de tiempo ni lugar.

3. Marco Teórico

3.1 Experiencia de Usuario

La UX incluye todas las emociones, creencias, preferencias, percepciones, respuestas, comportamientos y logros de los usuarios, que suceden antes, durante y después del uso de un producto [7]. De acuerdo con la norma ISO 9241-210, UX se puede definir de la siguiente manera: “las percepciones y respuestas de la persona resultantes del uso y / o uso anticipado de un producto, sistema o servicio” [8]. Establece que UX “incluye todas las emociones, creencias, preferencias, percepciones, respuestas físicas y psicológicas, comportamientos y logros de los usuarios que ocurren antes, durante y después del uso”. Además, la norma ISO 9241-210 señala que UX “es una consecuencia de la imagen de marca, la presentación, la funcionalidad, el rendimiento del sistema, el comportamiento interactivo y las capacidades de asistencia del sistema interactivo, el estado interno y físico del usuario resultante de una experiencia previa, experiencias, actitudes, habilidades, personalidad y el contexto de uso” [7].

3.2 Evaluación UX

Evaluar la UX de una aplicación es complicado, ya que depende de la expresión de la experiencia por parte de los usuarios [9]. Por lo tanto, además de medidas objetivas como el tiempo de ejecución de la tarea o el número de errores, es necesario desarrollar métodos de evaluación que indiquen las necesidades de satisfacción del usuario [10]. Para ayudar a la observación de UX, el autor Desmet sugiere el uso de dos tipos de instrumentos: no verbales (objetivos) y verbales (subjetivos). Los instrumentos no

verbales comprenden mecanismos que permiten observar el componente expresivo o fisiológico de una respuesta emocional. Por otro lado, los instrumentos de autoinforme verbal suelen evaluar el componente de sentimiento subjetivo de las emociones. La UX puede ser evaluada con diferentes métodos que permiten una apreciación emocional del usuario frente al producto [11].

3.3 Detección de Emociones

Se han realizado diversas investigaciones para reconocer las emociones e interpretarlas con múltiples fines, entre ellos la educación, salud, seguridad y marketing [12] [13] [14][15] [16]. La mayor parte de esos métodos se centran en indagar, observar y analizar las emociones generadas en los usuarios [17] [18]. Es posible reconocer emociones por diferentes modos o canales (multi-modalidad):

- La voz: por un micrófono, reconociendo modulación, tono, volumen.
- Las expresiones faciales o corporales: por medio de cámaras o sensores.
- Entradas del usuario: desde el texto escrito, cuando usa emoticones, signos de admiración, etc.
- Señales biológicas: con sensores que tomen temperatura, resistencia galvánica de la piel, electroencefalograma, pulso, etc.

4. Propuesta

Este trabajo propone aplicar una técnica automática, no invasiva para el usuario, que mediante detección facial de emociones permita evaluar la satisfacción de los usuarios mientras utilizan un sistema (Fig. 1). Con ello dejar un registro de las emociones asociadas a las características principales que se encuentran presentes en todo sistema web, permitiendo identificar las que necesiten mejoras a futuro.

El objetivo de esta investigación es “Diseñar y validar una evaluación automática de UX aplicando técnicas de detección facial para sistemas web, haciendo uso de JavaScript para la detección y el reconocimiento de rostros en el navegador implementada sobre la API principal de tensorflow.js y el uso de node, react, y base de datos no relacionales, que al trabajar en conjunto brinde información visual para apoyar a la toma de decisiones y de esta forma mejorar dicho sistema”.

4.1 Detalles de los elementos que intervienen en el sistema propuesto

Grupo de usuarios: grupo de personas que usarán la aplicación, de diferentes edades y género, a los que se les analizarán sus emociones faciales.

Tareas: conjunto de imágenes de algún sistema que se desee evaluar, dichas imágenes son presentadas con alguna pregunta o descripción sobre la imagen, luego de unos segundos aparecerá un cuestionario interactivo emocional que representa las emociones, el usuario deberá seleccionar alguna de estas para comunicar cómo se sintió al realizar esa tarea, con el fin de cotejarlas con las detectadas a través de la cámara.

Sistema web: es el sistema en general que se diseñará utilizando tecnología web, y disponible en un servidor, la pantalla principal al ingresar será un formulario, en el cual se pedirá datos como su nombre ficticio, género, y otros datos, que no comprometen la identidad del usuario ya que será totalmente anónima. Luego de esto, el usuario podrá elegir de una lista precargada qué sistema quiere analizar o realizar el test.

WebCam: mediante la cámara del dispositivo, en conjunto con la librería face api implementada sobre la API principal de tensorflow.js que utilizará para la detección facial.

Base de datos de registros de emociones: se almacena la información recolectada (emociones, y respuesta obtenida en la encuesta interactiva emocional) de los usuarios para cada test que realizó.

Procesamiento y clasificación de emociones: con los registros obtenidos se realizará una clasificación y análisis de resultados, para luego mostrar en gráficos, con el fin de dar una mejor visualización.

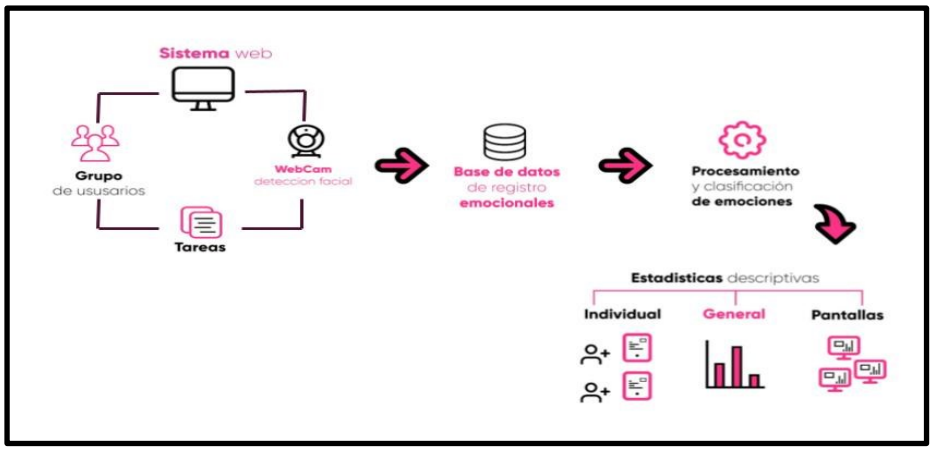


Fig. 1. Modelo preliminar de la propuesta (diseño de los autores).

5. Resultados esperados

Se espera obtener una evaluación de forma automatizada de Experiencia de Usuario (UX) con detección facial para sistemas web, usando un factor muy relevante como son las emociones de los usuarios. Contribuyendo con una técnica para la recolección de emociones provocadas durante su utilización.

Se pretende dejar a la comunidad del área, una técnica fiable, segura, no invasiva para el usuario, económica y además que se pueda realizar sin restricciones de tiempo ni lugar. Por la cual, mediante detección facial permite evaluar la satisfacción de los usuarios mientras utilizan un sistema. A su vez, dejar un registro de las emociones asociadas a las características principales que se encuentran presentes en todo sistema web, permitiendo identificar las que necesiten mejoras a futuro.

6. Trabajo futuro

Si bien esta propuesta de Trabajo Final de Grado, propone aplicar una técnica automática, no invasiva para el usuario, que mediante detección facial de emociones permita evaluar la satisfacción de los usuarios mientras utilizan un sistema, se podría desprender del presente estudio, considerar la propuesta para ser aplicada a una sala con varios usuarios o espectadores observando una sola pantalla o medio digital, con el fin de obtener un análisis y reporte por grupos.

Referencias

1. A. P. O. S. Vermeeren, E. L.-C. Law, V. Roto, M. Obrist, J. Hoonhout, and K. Väänänen-Vainio-Mattila, "User experience evaluation methods," in Proceedings of the 6th Nordic Conference on Human-Computer Interaction Extending Boundaries - NordiCHI '10, 2010, p. 521.
2. S. Kujala, V. Roto, K. Väänänen-Vainio-Mattila, E. Karapanos, and A. Sinnelä, "UX Curve: A method for evaluating long-term user experience," Interact. Comput., vol. 23, no. 5, pp. 473–483, Sep. 2011.
3. A. H. Allam, A. Razak, C. Hussin, and H. M. Dahlan, "User Experience: Challenges and Opportunities."
4. J. Vanattenhoven and D. De Roeck, "Investigating the Use of the Experience Clip Method," Int. Conf. User Centric Media, vol. 40, no. June, pp. 79–86, 2009.
5. P. Ekman, Emotions revealed: recognizing faces and feelings to improve communication and emotional life. Henry Holt and Co, 2004.
6. J. A. Russell, "Pancultural aspects of the human conceptual organization of emotions.," J. Pers. Soc. Psychol., vol. 45, no. 6, pp. 1281–1288, 1983.
7. "ISO 9241-11:2018(en), Ergonomics of human-system interaction — Part 11: Usability: Definitions and concepts," 2018.
8. "ISO 9241-11:1998 - Ergonomic requirements for office work with visual display terminals (VDTs) -- Part 11: Guidance on usability." [Online]. Available: https://www.iso.org/standard/16883.html. [Accessed: 03-Aug-2017].
9. S. Kujala, V. Roto, K. Väänänen-Vainio-Mattila, E. Karapanos, and A. Sinnelä, "UX Curve: A method for evaluating long-term user experience," Interact. Comput., vol. 23, no. 5, pp. 473–483, Sep. 2011.
10. A. P. O. S. Vermeeren, E. L.-C. Law, V. Roto, M. Obrist, J. Hoonhout, and K. Väänänen-Vainio-Mattila, "User experience evaluation methods," in Proceedings of the 6th Nordic Conference on Human-Computer Interaction Extending Boundaries - NordiCHI '10, 2010, p. 521.
11. P. M. Desmet, "Measuring emotions: Development of an instrument to measure emotional responses to products," Funology from usability to enjoyment. Kluwer Acad. Publ. Dordrecht, Boston, Londres, no. Blythe, M. A.; Overbeeke, K.; Monk, A. F.; Wright, P. C. (ed.), 2003.
12. I. Zualkernan, F. Aloul, S. Shapsough, A. Hesham, and Y. El-Khorzaty, "Emotion recognition using mobile phones," Comput. Electr. Eng., vol. 60, pp. 1–13, 2017.
13. X. Wang, X. Chen, and C. Cao, "Human emotion recognition by optimally fusing facial expression and speech feature," Signal Process. Image Commun., vol. 84, no. March, p. 115831, 2020.
14. A. Vartanov, V. Ivanov, and I. Vartanova, "Facial expressions and subjective assessments of emotions," Cogn. Syst. Res., vol. 59, no. 2020, pp. 319–328, 2020.

15. S. Patil, A. Joshi, G. Deore, A. Taley, and S. Sawant, “Emotion Detection Using Facial Recognition,” SSRN Electron. J., 2020.
16. L. N. Aballay, S. V. Aciar, and C. A. Collazos, “Emotions for Virtual Learning Environments,” IEEE-RITA, vol. 9, no. 3, 2021.
17. L. Aballay, Al. Orellana, S. Aciar, and C. Collazos, “Affective Evaluation: Proposal for Educational Platforms,” in Jornadas Internacionales de Campus Virtuales.
18. L. Aballay, C. Collazos, and S. Aciar, “Emotions evaluation on Virtual Learning Environments: Students perceptions,” in VII Jornadas Ibero-americanas de Interação Humano-Computador - HCI 2021, 2021.

I+IoT RLab
Laboratorio Remoto de IoT e IIoT, para la enseñanza, la asistencia y consultoría a empresas

Gabriel Ambrosio[1], Mariano Zapata[1], Gustavo Mercado[2], Carlos Taffernaberry[2], Ana Laura Diedrichs[2], Sebastián Tobar[2], Cristian Pérez Monte[2], Ana Lattuca[3], Nelson Merino[3], Néstor Manzur[3], Antonio Álvarez[3]

Ingeniería Electrónica, Facultad Regional Mendoza, UTN ambrosiogabrielgerman@gmail.com, marianozapata10@gmail.com

gridTICs, Ingeniería Electrónica, Facultad Regional Mendoza, UTN

{gmercado, cartaffe, ana.diedrichs, mstobar,cfperez}@frm.utn.edu.ar

Cátedra Proyecto Final, Ingeniería Electrónica, Facultad Regional Mendoza, UTN ana.lattuca@docentes.frm.utn.edu.ar, nelson.merino@frm.utn.edu.ar, nestor.manzur@gmail.com, antonioalvarezabril@yahoo.com.ar

Abstract. Las redes de Internet de las Cosas (IoT) e Internet de las Cosas Industriales (IIoT) han tenido un enorme desarrollo en los últimos años, debido a su amplio campo de uso, amplia variedad y simplicidad en su desarrollo. En el presente trabajo se propone el desarrollo de un laboratorio remoto para poder enseñar, estudiar y desarrollar las técnicas, arquitecturas y protocolos de estos dispositivos. En el desarrollo se hará uso de infraestructura virtualizada para la administración de reservas/recursos de forma dinámica, una computadora embebida Raspberry Pi para la comunicación entre dicha infraestructura y los dispositivos finales, identificados como Motes.

Keywords: IoT · IIoT · Raspberry Pi · 6LoWPAN · Contiki · OpenMote · Infraestructura · Proxmox

Contexto

El presente trabajo está inserto en el Proyecto Final de Grado denominado "Laboratorio Remoto de IoT e IIoT, para la enseñanza, la asistencia y consultoría a empresas", acreditado por la Cátedra Proyecto Final de la Carrera de Ingeniería de Electrónica de la Universidad Tecnológica Nacional.

El trabajo final es llevado adelante por alumnos tesistas y con la supervisión de docentes de la cátedra. Mientras que el grupo gridTICs, que cumple funciones de stakeholder, tiene las tareas de dirección técnica, asesoramiento, financiamiento y provisión de laboratorios.

1. Introducción

El concepto "Internet de las Cosas"[1] (IoT, por sus siglas en inglés: Internet of Things), se refiere a la comunicación entre dispositivos sin la intervención directa de los seres humanos. En resumen, IoT consiste en la integración de dispositivos

electrónicos en objetos de uso cotidiano, permitiendo que se conecten a Internet, con el objetivo de lograr el acceso remoto a los datos generados, o el accionamiento de actuadores a distancia.

Estos son implementados con microcontroladores que tienen un bajo poder de cómputo, generalmente poseen comunicación inalámbrica pero no están restringidos a esta y sensores que permiten la medición de múltiples parámetros de acuerdo al ambiente en el que se encuentren instalado, entre los más conocidos se puede identificar humedad, temperatura, posición, pero también puede ser corrosión producida en una tubería de gas, concentración de gases tóxicos en tanques de petróleo, niveles de tensión, corriente y potencia en líneas de distribución de energía, entre otros.

El presente proyecto tiene como objetivo proporcionar a usuarios, universidades y pequeñas empresas interesadas en IoT/IIoT, un entorno para estudiar y enseñar técnicas, arquitecturas y protocolos de dispositivos IoT/IIoT.

En el desarrollo de hardware se utilizarán los sensores OpenMote cc2538 [2] como los nodos sensores de múltiples dispositivos de medición conectados de forma alámbrica a estos. El sistema operativo Contiki [3] será utilizado en los nodos, ya que cuenta con ventajas como el stack completo TCP/IP y compatibilidad con 6LoWPAN [4].

Los nodos sensores se conectarán a una computadora mediante un hub USB, lo que facilitará el acceso a los mismos mediante una terminal de línea de comandos. Esto permitirá a los usuarios programar cada nodo de forma independiente, brindando flexibilidad y control en el proceso de desarrollo y configuración.

Los múltiples nodos sensores serán enlazados de forma inalámbrica a un nodo coordinador (Gateway) que estará conectado a una placa Raspberry Pi. La comunicación entre los nodos utilizará el estándar IEEE 802.15.4 [5] y el protocolo 6LoWPAN para permitir la comunicación mediante IPv6, especialmente útil en entornos industriales donde el número de dispositivos asciende a miles o ciento de miles. Los datos obtenidos de los sensores se almacenarán en una base de datos, aunque el tipo específico de base de datos no ha sido decidido aún.

El nodo coordinador, funcionando como Gateway, se conectará mediante USB a una placa Raspberry Pi [6] seleccionada según los lineamientos del proyecto. Esta utilizará el sistema operativo Raspbian-OS, optimizado para su hardware, y será el intermediario para el intercambio de información con la infraestructura virtualizada posteriormente mencionada. La Raspberry Pi se encargará de programar el nodo coordinador y actuará como conector entre la computadora mencionada anteriormente y el propio nodo coordinador. Es decir que la configuración de los diferentes nodos coordinadores será realizada mediante interfaz USB o ethernet, pero la comunicación entre este y los nodos sensores será realizada mediante el estándar 6LoWPAN.

El resto de los componentes y sistemas se desplegarán utilizando el entorno de virtualización Proxmox™ [7]. Se ejecutarán máquinas virtuales, contenedores y otros recursos para administrar el uso de los nodos según las reservas realizadas por los usuarios.

La administración de conexiones remotas se realizará mediante un controlador de dominio con autenticación previa. Se requerirá la creación de usuarios con permisos adecuados para acceder a los recursos.

Finalmente se utilizará una herramienta libre y de código abierto que permita el acceso remoto al equipamiento utilizando únicamente un navegador web, sin la necesidad de ninguna aplicación cliente.

En la siguiente etapa se explica con mayor detalle los componentes mencionados anteriormente.

2. Componentes del Sistema

Se enumeran y describen a continuación todos los componentes utilizados para constituir el sistema completo del Laboratorio Remoto de IoT.

2.1 Red de sensores Inalámbricos

El diseño de la red se fundamenta en el uso de nodos sensores OpenMote CC2538, que estarán equipados con una variedad de sensores como temperatura, humedad y luz, entre otros. Cuentan con zócalos adaptadores para conectarse mediante USB o ser alimentados por baterías.

Para establecer la comunicación entre los nodos, se empleará el estándar IEEE 802.15.4, que se encarga de definir el nivel físico y el control de acceso al medio para redes inalámbricas de área personal. Este estándar es ideal para redes que necesitan transmitir datos a baja velocidad, ya que está diseñado para priorizar el ahorro máximo de energía en su funcionamiento. La figura 1 muestra los nodos sensor y coordinador.



Figura 1. Nodo sensor (izquierda) y nodo coordinador (derecha).

2.2 Nodo Coordinador o Router de Borde

Los nodos OpenMoe ofrecen la posibilidad de configurar uno de ellos con un adaptador USB para que funcione como un router de borde. Esto les permite soportar una topología de red en malla, donde cada nodo está interconectado con uno o más nodos de la red. Esta configuración en malla reduce el riesgo de fallos en la red y, como resultado, disminuye la necesidad de mantenimiento periódico.

2.3 Gateway

El dispositivo se conecta al coordinador mediante una interfaz USB y a Internet (IPv6) a través de una interfaz Ethernet o WiFi. Su función principal es realizar la adaptación del protocolo IPv6 de Internet para que pueda ser utilizado en la capa de enlace de la red de sensores (802.15.4).

El gateway se implementará en una placa Raspberry Pi, específicamente en la Raspberry Pi 3 B+. Esta placa es una computadora de bajo costo y tamaño reducido, aproximadamente del tamaño de una tarjeta de crédito, diseñada con el propósito de fomentar la enseñanza de la informática en las escuelas. Está equipada con varios componentes esenciales para el funcionamiento de una computadora convencional y puede comportarse como tal.

Para este proyecto, se ha seleccionado la Raspberry Pi 3 B+, que ofrece las siguientes características: un procesador Cortex-A53 (ARMv8) de 64 bits con una velocidad de 1,4 GHz, 1 GB de RAM, conectividad LAN inalámbrica IEEE 802.11 de 2.4 GHz y 5 GHz, Bluetooth 4.2, Gigabit Ethernet, salida de video HDMI, 4 puertos USB 2.0 y una ranura para tarjetas micro SD utilizada para cargar el sistema operativo y almacenar datos, entre otras características.



Figura 2. Raspberry Pi con nodo coordinador o router de borde.

2.4 Almacenamiento de datos

Cada vez que la aplicación cliente de datos obtiene información de los nodos sensores, se guardará en una base de datos.

2.5 Visualización

El servidor tiene la capacidad de permitir la visualización y el control de los datos recopilados por los nodos sensores, así como también la programación de los tiempos de adquisición de datos y las alertas. Además, facilita la realización de comparaciones entre las diferentes medidas.

Para acceder a la funcionalidad de visualización como cliente, se utiliza un navegador o browser

2.6 Gestor de usuarios y máquinas virtuales

Los recursos demandados por los entornos de desarrollo serán esencialmente computacionales. Por este motivo, será conveniente la utilización de Apache VCL (Virtual Computing Lab) [8] para la gestión de los entornos, como se muestra en la figura 3. Apache VCL es una plataforma de computación en la nube libre y de código abierto cuyo objetivo principal es brindar entornos informáticos personalizados a los usuarios. Estos entornos podrán ser algo tan sencillo como una máquina virtual ejecutando software de oficina o algo tan poderoso como un clúster corriendo complejas simulaciones. Para el caso particular de este proyecto, Apache VCL gestionará los entornos de desarrollo de los módulos de IoT. La interfaz de usuario consistirá en un autoservicio web. A través de este portal, los usuarios podrán seleccionar los ambientes y hacer reservas en el futuro.

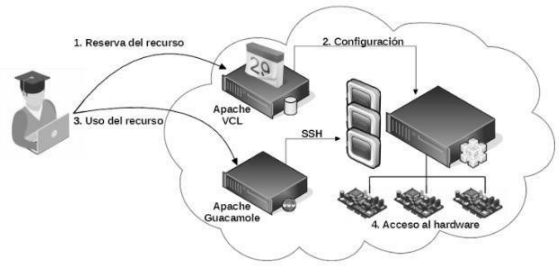


Figura 3. Gestión de usuarios y máquinas virtuales.

2.7 Acceso Remoto

El entorno de desarrollo implementado en el proyecto será accesible a través de emuladores de terminal en el futuro. Con el objetivo de simplificar el acceso al laboratorio, se optará por la interfaz de acceso web Apache Guacamole [9], que brinda emuladores de terminal dentro del navegador, sin requerir ningún software adicional.

Referencias

1. Waher, P.: Learning Internet of Things, 1st Edition. Packt Publishing, UK (2015).
2. OpenMote-cc2538. Disponible en https://doc.riotos.org/group__boards__openmotecc2538.html. Julio 2023.
3. Contiki: The Open Source OS for the Internet of Things. Disponible en https://www.contikios.org. Mayo 2019.
4. Kushalnagar, N., Montenegro, G., Schumacher, C.: IPv6 over low-power wireless personal area networks (6LoWPANs): overview, assumptions, problem statement, and goals (2007).
5. Molisch, A.F., Balakrishnan, K., Chong, C.C., Emami, S., Fort, A., Karedal, J., Kunisch, J., Schantz, H., Schuster, U. and Siwiak, K.,: IEEE 802.15. 4a channel model-final report. IEEE P802, 15(04), p.0662 (2004).
6. Raspberry Pi. Disponible en https://www.raspberrypi.org. Julio 2023.
7. Proxmox. Disponible en https://www.proxmox.com/en/. Julio 2023.

8. Apache VCL. Disponible en https://vcl.apache.org/ Julio 2023.
9. Apache Guacamole. Disponible en https://guacamole.apache.org/releases/1.5.2/ Julio 2023.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Reconocimiento de Logotipos de Empresas Tecnológicas mediante Redes Neuronales Artificiales

Ignacio Ibañez[1], Juan Bautista Rodriguez[1], Cristian Desiderio Arana[1], Cinthia Vegega[1,2], María F. Pollo-Cattaneo[1,2]

[1] Universidad Tecnológica Nacional. Facultad Regional La Plata. Argentina
[2] GEMIS. Universidad Tecnológica Nacional. Facultad Regional Buenos Aires. Argentina
ignacio@innovotecnologias.com.ar, cinthia.vegega@gmail.com, flo.pollo@gmail.com

Resumen. En el presente trabajo, se detallan y comparan los resultados obtenidos de la aplicación de dos arquitecturas posibles de Redes Neuronales Artificiales, con el fin de reconocer logotipos de empresas del sector tecnológico, en el marco de un trabajo de cátedra de la asignatura Inteligencia Artificial de la carrera de Ingeniería en Sistemas de Información de la Universidad Tecnológica Nacional, Facultad Regional La Plata.

Palabras Clave: Inteligencia Artificial, Redes Neuronales Artificiales, Redes Convolucionales, Redes Bakpropagation

1. Introducción

Dentro del ámbito de la Universidad Tecnológica Nacional, Facultad Regional La Plata, la asignatura Inteligencia Artificial dictada en el quinto año de la carrera de Ingeniería en Sistemas de Información tiene como uno de sus objetivos que el estudiante logre analizar, formalizar e implementar problemas a través de la implementación de Redes Neuronales Artificiales, aplicando metodologías, técnicas y herramientas pertenecientes a las buenas prácticas profesionales [1]. Es a partir de este objetivo que las docentes de la cátedra, María Florencia Pollo-Cattaneo y Cinthia Vegega (que pertenecen al grupo de investigación GEMIS [2]), proponen a los estudiantes que transitan esta materia, la realización de un proyecto que pueda ser resuelto utilizando dicha tecnología. En este contexto, un grupo de estudiantes propone para su trabajo, la comparación de dos tipos de Redes Neuronales Artificiales con el objetivo de abordar la problemática del reconocimiento de logotipos de empresas del rubro tecnológico ya que la aplicación de software tradicional para este tipo de tarea resultaría laboriosa y poco eficiente [3]. En primera instancia se describen los elementos de trabajo (sección 2), para luego mostrar los resultados obtenidos (sección 3). Finalmente, se presentan las conclusiones y futuras líneas de trabajo (sección 4).

2. Elementos de Trabajo

El objetivo final del trabajo es obtener dos modelos predictivos capaces de reconocer y distinguir el logo de cada una de las siguientes empresas del rubro tecnológico: AMD, Apple, Cisco, IBM y Microsoft. La situación ideal es que sean capaces de reconocer el logo dispuesto de distintas formas y con sus respectivos

cambios a lo largo de los años. Generalmente, la aproximación a este tipo de casos consiste en aplicar Redes Neuronales Convolucionales (CNN) [4] en dos vertientes: Logo Detection (LD), a fin de determinar la posición del logo, y Logo Recognition (LR), que trata de identificar a qué empresa pertenece dicho logo. En este caso, se analiza la última vertiente. Además, se considera conveniente utilizar dos enfoques de trabajo distintos, y luego comparar los resultados obtenidos a través de cada uno de ellos:

- *Red Neuronal Artificial Convolucional realizada en Python con librería externa*
- *Red Neuronal Artificial Feedforward con Backpropagation desarrollada en C#*

Para ello, se utiliza un conjunto de imágenes de prueba y entrenamiento de un repositorio público de GitHub "Logo-2k-plus-Dataset" [5].

El primer enfoque consiste en la utilización de una Red Neuronal Artificial (RNA) del tipo convolucional. Se trata de una variante de las redes perceptrón multicapa (o multi-perceptrón), y se las considera ideales para aquellas tareas que involucren "visión artificial" como, por ejemplo, la clasificación y segmentación de imágenes, dado que "imitan" el comportamiento de las neuronas que componen la corteza visual primaria en un cerebro humano. Se utiliza Python como lenguaje de programación, en conjunto con las librerías *TensorFlow* [6] y *Keras* [7]. Ambas de código abierto, ampliamente utilizadas y difundidas en el campo del aprendizaje automático y de la construcción de redes neuronales. La topología se conforma por 4 capas. La primera se denomina *Flatten* o capa de aplanamiento. Se utiliza para convertir los datos de entrada (las imágenes) en un vector unidimensional. La segunda es *Dense (1/2)*. Está totalmente conectada y se compone de 128 neuronas. Su función de activación es ReLu (unidad lineal rectificada). La tercera es *Dropout***.** Si bien se considera una capa, el "dropout" o "capa de abandono" es, en realidad, una técnica de regularización utilizada para reducir el sobreajuste en la red neuronal. Particularmente para esta red, se estableció en un valor de 0,2 de las entradas a cero de forma aleatoria en cada paso de actualización. La última es *Dense (2/2)* que es la capa de salida, también totalmente conectada y compuesta únicamente de 5 neuronas (correspondientes a cada una de las posibles clasificaciones / empresas). La función de activación es Softmax, y permite estimar la probabilidad de que la entrada pertenezca a cada clase [8]. Por último, cabe mencionar que se selecciona como optimizador al *algoritmo de Adam*, basado en gradientes, y *Categorical Cross Entropy* como función de pérdida.

El segundo enfoque, sin vinculación con librerías de terceros específicas, se logra al ejecutar un experimento en el entorno de desarrollo de software Microsoft .NET [9, 10] conformado por una solución compuesta de 3 proyectos individuales. El primero, es una aplicación de escritorio capaz de consultar imágenes en la web, para luego descargarlas, redimensionarlas y ubicarlas en una estructura de directorios adecuada para el entrenamiento y prueba de la RNA El segundo, es un proyecto de tipo *biblioteca* denominado RNA que contiene las clases que constituyen la red neuronal en sí. Puede utilizarse desde diferentes tipos de proyectos y plataformas presentando una RNA con topología de 3 capas (entrada, oculta y salida). La RNA es del tipo *feed-forward* (todas sus conexiones son hacia adelante, no hay conexiones hacia atrás o recursivas) con aprendizaje mediante el algoritmo *backpropagation*. Este algoritmo es del tipo "corrección del error por propagación hacia atrás" (del inglés *backpropagating error correction*). Se implementa mediante optimización iterativa de primer orden en lo que

se conoce como “descenso del gradiente”. Permite encontrar mínimos locales de una función diferenciable (las de activación de la RNA). La interpretación del gradiente es fundamental: indica la dirección de máximo crecimiento de la función. Por lo tanto, si se buscan mínimos locales, se debe tomar la misma dirección del gradiente en el punto analizado, pero en sentido contrario. La cantidad de neuronas de entrada está íntimamente relacionada con la cantidad de pixeles que constituyen la imagen. Así, cuando se trabaja con imágenes de 200 pixeles, se generan 40 mil neuronas de entrada (200 * 200). La capa de salida tiene una neurona por cada resultado posible. Al evaluar 5 logos de empresas, se cuenta con 5 neuronas de salida. Al solicitarle a la RNA que infiera el nombre de un logo presentado, las 5 neuronas resultan en un valor comprendido entre 0 y 1. Un valor de 1 indica que esa neurona presenta “certeza”. El último, es otro proyecto de ventanas de escritorio que permite entrenar, probar y hacer uso de la RNA para la identificación de logos. Este último proyecto se denomina App y referencia al proyecto RNA antes mencionado para hacer uso de la red neuronal. Presenta una interfaz gráfica (figura 1) capaz de establecer la función de activación (Sigmoide, Tangente hiperbólica, ReLU o Leaky ReLU), la cantidad de neuronas en la capa oculta, las iteraciones (*epochs*) y la tasa de aprendizaje inicial. Por último, presenta un control que permite activar la recreación de la RNA en cada entrenamiento. Para monitorear el funcionamiento de la misma se presenta el elemento (la empresa) de la cual se están procesando sus logos y debajo del nombre de la empresa, se visualiza la imagen identificada. Sobre la derecha, la tasa de aprendizaje y la exactitud. Cuanto menor sea el valor de la tasa de aprendizaje, más lento se realizará ese proceso. En el caso de la exactitud, es óptima si vale 1. A fin de lograr rigurosidad en el lenguaje, se aclara que se utiliza genéricamente el término logo para referirse a logotipos, imagotipos (texto e imagen), isotipos o isologos.



Fig. 1. Interfaz Proyecto App.

3. Resultados

Referido a la **RNA desarrollada en Python**, se observa que el error general obtenido durante la fase de entrenamiento oscila en cada corrida del código. Sin embargo, tras varias ejecuciones, el entrenamiento resulta en 84,36% de precisión para 35 épocas. Si bien no se trata de una muestra realmente significativa para obtener conclusiones definitivas, preliminarmente se observa que la precisión obtenida (80%), es similar a la que arrojó la librería *Keras* durante la etapa de entrenamiento. Para obtener una aproximación más rigurosa, con un volumen mayor de corridas, a través de un sencillo algoritmo se toma el 25,6% de las imágenes disponibles en los conjuntos, para validar la predicción de la red entrenada (22 de AMD, 18 de Apple, 14 de Cisco,

22 de IBM y 24 de Microsoft, totalizando 100 imágenes). En este caso, los resultados obtenidos son los siguientes: 91 validaciones correctas y 9 incorrectas. El error es del 9%, aún menor que el error general calculado durante el entrenamiento.

Referido a la **RNA desarrollada en C#**, se alcanza una exactitud máxima de 83% mediante la función de activación Sigmoide, 50 neuronas en la capa oculta y una tasa de aprendizaje de 0,0001. Este valor se logró con aproximadamente 1.500 *epochs*. Este enfoque presenta varias ventajas y desventajas que se enuncian a continuación, entendiéndose, tácitamente, que aquellas que resulten una ventaja de una de las opciones será una desventaja de la otra.

Ventajas

- El control sobre los algoritmos en ejecución es total.
- No hay costos potenciales de, por ejemplo, licencias que los desarrolladores de la librería puedan establecer a futuro.
- El ciclo de vida del *software* no depende de terceros.
- Puede registrarse la propiedad intelectual (puede haber restricciones, pero ciertamente menos que con una librería externa).
- La seguridad es íntegramente controlada por la organización o particular propietario. Es importante destacar aquí, que ante una solución de código abierto como es TensorFlow, podríamos analizar el código en busca de brechas de seguridad. Luego de pasar exitosamente el análisis podríamos compilarla en infraestructura propia para lograr un alto grado de seguridad.
- Desde el punto de vista conceptual, resulta más provechosa dado que expone la estructura interna y sus pilares conceptuales fundamentales mucho más explícitamente. Esto puede facilitar el ajuste de los parámetros para mejores los resultados.
- Debe respaldarse en bibliografía teórica que puede involucrar conceptos de otras áreas de conocimiento. En contrapartida, las librerías externas con cierto grado de maduración ponen a disposición manuales de uso que suelen ser de rápido aprovechamiento.

Desventajas

- Se requieren integrantes en el equipo de desarrollo con un nivel de experiencia medio o alto en el marco de trabajo seleccionado (.NET en este caso). Por el contrario, el uso de librería externas suele favorecer a los programadores inexpertos (desde uno o dos años de experiencia en la actividad).
- La curva de aprendizaje resulta en mayor tiempo insumido para ser capaces de aportar mejoras y disponer de un prototipo funcional con resultados aceptables.
- Desarrollar una solución propia conlleva cierta falta de certeza al momento de escalar en requerimientos. Las librerías de terceros desarrolladas por grandes comunidades (como TensorFlow de Google) brindan un respaldo fundamental en este aspecto y se nutren de aportes constantes de su comunidad. Además, no es

posible variar con facilidad la topología, sus funciones de activación y demás parámetros.

Finalmente, se concluye que ambos enfoques propuestos ofrecen resultados acordes, cada uno con sus fortalezas y debilidades.

4. Conclusiones

El objetivo de este trabajo consiste en reconocer logotipos de empresas del sector tecnológico utilizando para ello dos enfoques: una Red Neuronal Artificial Convolucional realizada en Python con librería externa y una Red Neuronal Artificial Feedforward con Backpropagation desarrollada en C#. Los resultados obtenidos con ambos enfoques fueron exitosos para el objetivo planteado inicialmente. La red neuronal convolucional desarrollada en Python presenta un rendimiento aceptable al retornar más del 80% de aciertos. Por otro lado, la red desarrollada en C# también alcanza una exactitud de más del 80%. Como futuras líneas de trabajo, dado que el dataset propuesto para las pruebas y entrenamiento es relativamente pequeño, se podría ampliar ajustando, además, algunos parámetros para obtener un error menor en cada una de las redes utilizadas. Asimismo, este trabajo se presentará como ejemplo de aplicación el cuatrimestre que viene, dejándolo disponible para que los estudiantes que cursan la asignatura de Inteligencia Artificial puedan utilizarlo como apoyo para sus respectivos trabajos de cátedra.

Referencias

1. Universidad Tecnológica Nacional, Facultad Regional La Plata (2023). Programa Analítico de la asignatura Inteligencia Artificial. Disponible en: https://www.frlp.utn.edu.ar/sites/default/files/inteligenci_artificial_1150.pdf Último Acceso: 29/07/2023.
2. Grupo GEMIS (2023). Historia Grupo GEMIS. Disponible en http://grupogemis.com.ar Último Acceso: 29/07/2023.
3. Nielsen, M. A. (2018). Neural Networks and Deep Learning. Neuralnetworksanddeeplearning.com; Determination Press. Disponible en: http://neuralnetworksanddeeplearning.com/chap1.html Último Acceso: 29/07/2023.
4. Raúl Castilla Bravo. Universidad Europea de Madrid. Reconocimiento de logotipos de marcas mediante Redes Neuronales de Convolución (CNN). https://titula.universidadeuropea.com/bitstream/handle/20.500.12880/767/CastillaBravoRaul.pdf Último Acceso: 29/07/2023.
5. A Large-Scale Logo Dataset for Scalable Logo Classification. GitHub. https://github.com/msn199959/Logo-2k-plus-Dataset Último Acceso: 29/07/2023.
6. API Documentation de TensorFlow Core v2.1.0. (n.d.). www.tensorflow.org/api_docs Último Acceso: 29/07/2023.
7. Team, K. (n.d.). Keras documentation: Keras API reference. Keras.io. https://keras.io/api
8. Neural networks. (n.d.). Ml4a.github.io. https://ml4a.github.io/ml4a/neural_networks Último Acceso: 29/07/2023.
9. Gary, Miller. A 3-Layer Feed-Forward Neural Net in C#, with Graphical Display. https://www.codeproject.com/Articles/5272136/A-3-Layer-Feed-Forward-Neural-Net-InCsharp-with-G Último Acceso: 29/07/2023.

10. García Martínez, Servente & Pasquini (2007) Sistemas Inteligentes. Editorial Nueva Librería.

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Análisis Discriminante sobre Datos de Egresados Universitarios para la Toma de Decisiones Institucionales

Simón Pedro González [1], Myriam Herrera[1] y Susana Ruiz[2]

1 Instituto de Informática, Facultad de Ciencias Exactas Físicas y Naturales, San Juan, San Juan 5400, Argentina

2 Departamento de Matemática, Física y Química, Facultad de Ciencias Exactas Físicas y Naturales, San Juan, San Juan 5400, Argentina

simon.pedro.g@gmail.com, myriamhrrr@gmail.com, sbruizr@yahoo.com.ar

Resumen. En el presente trabajo se plantean técnicas de regresión logística y reducción de modelos para analizar datos de egresados de la Facultad de Ciencias Exactas, Físicas y Naturales de la Universidad Nacional de San Juan. El objetivo de los modelos implementados es determinar relaciones entre los observables y dos variables respuesta relacionadas con el futuro laboral inmediato de los egresados y el tiempo que les tomó obtener el título. Además, para comparar el desempeño, se generaron clasificadores utilizando redes neuronales. Los modelos desarrollados servirán para plantear herramientas de análisis para apoyar procesos de toma de decisiones institucionales.

Palabras clave: Egresado, Análisis Discriminante, Regresión Logística.

1. Introducción

Los egresados son una fuente valiosa de retroalimentación para la Universidad, ya que permiten conocer cómo se desempeñan los ex-alumnos en el ámbito laboral y cómo fueron percibidos distintos aspectos del cursado. Este tipo de análisis es fundamental a la hora de considerar decisiones institucionales.

En este trabajo, se proponen métodos del estado del arte, con algunas mejoras adicionales, para analizar datos de egresados. Además, se aplicaron a los resultados de la encuesta SIU-Kolla al recién graduado. Los datos utilizados pertenecen a las encuestas de los años 2014, 2015 y 2019, de la Facultad de Ciencias Exactas, Físicas y Naturales, de la Universidad Nacional de San Juan, que suman un total de 161 registros. Las variables activas se muestran en la Tabla 1 y las variables objetivo, en la Tabla 2. Las categorías de referencia se encuentran indexadas con el valor cero.

La variable objetivo “Condición Actual” indica si el estudiante está trabajando o no en relación con la profesión para la cual estudió al momento de completar la encuesta. Esta variable constituye un indicador de qué perfiles de egresados son capaces de obtener un puesto de trabajo a corto plazo. La segunda variable objetivo, “Tiempo de cursado”, permite estudiar qué características determinan las demoras en el cumplimiento de los plazos previstos para cada programa. La selección de las categorías para esta variable se fundamenta en que la mediana de lo que tardaron los egresados en titularse es el doble a la duración prevista para el programa universitario que cursaron.

Tabla 1. Variables activas.

Cód.	Variable	Categorías
B1	Sexo	0. Femenino, 1. Masculino
B2	Edad	Variable numérica
B3	Estado Civil	0. Soltero, 1. Unido
B4	Tiene Hijos	0. No, 1. Sí
B5	Grupo de Convivencia	0. Familia Parental, 1. Familia Propia, 2. Solo
H1	Hábito de lectura	0. Diario, 1. Semanal o Mensual
H2	Tipo de lectura	0. Entretenimiento, 1. Libros relacionados con la profesión, 2. Revistas científicas
H3	Otros hábitos	0. Actividad física, 1. Actividades solidarias, 2. Escuchar música, 3. Ver televisión o escuchar radio, 4. Otras
E1	Padre (nivel educativo)	0. Primarios o menos, 1. Secundarios completos, 2. Secundarios incompletos, 3. Superiores completos, 4. Superiores incompletos
E2	Madre (nivel educativo)	Ídem variable “Padre”.
P1	Los contenidos estudiados durante la carrera cumplieron sus expectativas	0. De acuerdo, 1. En desacuerdo, 2. Ni de acuerdo ni en desacuerdo
P2	Nivel de exigencia de la carrera	0. Alto, 1. Bajo, 2. Medio
P3	Perfil del plan de estudios	0. Debería ser más generalista, 1. Debería ser más técnico, 2. Perfil adecuado
P4	Extensión del plan de estudios	0. Debería ser más corto, 1. Debería ser más largo, 2. Extensión adecuada
M1	Motivos por los cuales eligió su carrera universitaria	0. Desarrollo profesional, 1. Deseos de ser útil a la sociedad, 2. Interés por el avance científico y tecnológico, 3. Posibilidades de trabajo, 4. Una clara vocación, 5. Otros
C1	Conocimiento de Inglés	0. Básico, 1. Bueno
C2	Conocimiento de Herramientas Informáticas	0. Básico, 1. Bueno
C3	Características que cree son valoradas por los empleadores	0. La capacidad de adecuarse a los fines de la empresa, 1. La capacidad de resolver problemas, 2. La capacidad de trabajar en equipo, 3. La flexibilidad y adaptación al cambio, 4. La utilización de técnicas informáticas, 5. Las habilidades adquiridas, 6. Otras

Tabla 2. Variables respuesta.

Cod.	Variables	Categorías
T1	Tiempo de cursado	0. Mayor al doble de la duración pevista, 1. Menor al doble de la duración prevista
T2	Condición Actual	0. No trabaja en relación a la profesión, 1. Trabaja en relación a la profesión

2. Metodología

El modelo Logit, que consiste fundamentalmente en regresión lineal con una transformación logística, resulta útil en este caso debido a que es aplicable a conjuntos de datos multivariantes no normales con variables categóricas.

En general, al aplicar regresión logística, se busca obtener el modelo más simple que se ajuste correctamente a los datos, para una fácil interpretación. Existen distintos criterios de simplificación, siendo uno de los más utilizados el criterio de información de Akaike [1]. Sin embargo, al utilizarlo, es posible que el modelo resultante incluya variables que no son significativas en términos de su contribución a la disminución de la desviación entre el modelo y los datos.

En este trabajo, se plantea una metodología que consiste en reducir el modelo saturado, en primer lugar utilizando el criterio de Akaike con la librería MASS [2], y luego, utilizando el criterio de la desviación [3], que será descrito a continuación. Sea y la variable respuesta, y p_i las probabilidades obtenidas mediante un modelo Logit con parámetros β, la desviación se define a partir de la función de verosimilitud de la siguiente manera:

$$L(\beta)=logP(\beta)=\sum_{i=1}^{n}\left(y_i\, logp_i+(1-y_i)\log\left(1-p_i\right)\right) \qquad D(\beta)=-2\,L\,D(\beta)\geq 0\; D(\beta)\sim\chi_k^2$$

La reducción por el criterio de la desviación consiste en determinar si es significativo incluir una variable o conjunto de ellas en la ecuación del modelo. Sean los parámetros del modelo:

$$\beta=(\beta_1\beta_2) \qquad |\beta|=p \qquad |\beta_2|=s \qquad |\beta_1|=p-s$$

Para determinar si los parámetros de β_2 son significativos se requiere contrastar la hipótesis $H_0{:}\beta_2=0$ $H_1{:}\beta_2\neq 0$. El rechazo de H_0 implica que los parámetros β_2 son significativos para el modelo, por lo tanto, no se puede prescindir de ellos. H_0 se puede contrastar de la siguiente manera:

$$D(H_0)=-2L(\widehat{\beta}_1)\; D(H_1)=-2L(\widehat{\beta}_1) \quad D(H_0)\geq D(H_1)$$
$$\Delta D=D(H_0)-D(H_1)=2\,L(\beta_1,\beta_2)-2\,L(\beta_1) \qquad \Delta D\sim\chi_s^2$$
$$Rechazar\; H_0=\Delta D>\chi_s^2(\alpha) \qquad \alpha=0.05$$

En este trabajo, se implementó un algoritmo de reducción basado en el criterio de la desviación con una heurística de eliminación hacia atrás. Esto quiere decir que el proceso comienza con el modelo saturado M_n y en cada paso, se analizan todas las posibilidades de eliminación, y se selecciona aquella que pasa más holgadamente el test, en términos de nivel de significación.

El conjunto de datos se dividió en 80% datos de entrenamiento y 20% datos de prueba. Para evaluar el desempeño, se utilizaron métricas de clasificación derivadas de la matriz de confusión [4], a saber, la precisión (PR), la exhaustividad (RC), el Valor F1 y el Coeficiente de Correlación de Matthews (MCC). También se utilizó el área bajo la curva ROC (AUC).

Finalmente, para comparar el desempeño logrado, se generaron modelos de redes neuronales que permiten representar relaciones no lineales. La arquitectura utilizada cuenta con un nodo de entrada por cada categoría, un único nodo de salida, 32 capas ocultas y una función de activación sigmoidea. Para el entrenamiento, se utilizó como función de costo la entropía binaria cruzada y ajuste por gradiente descendiente. El sobreajuste se evita utilizando el método de parada anticipada, es decir, se detiene el entrenamiento cuando no se evidencia mejoría en la precisión del modelo sobre los datos de prueba.

3. Resultados

A continuación se muestra el modelo saturado (M_n) y el modelo doblemente reducido (M_D) para el caso positivo de la variable "Tiempo de cursado", es decir, la categoría "Menor al doble de la duración prevista".

$$\begin{aligned} M_n: \ln\frac{p}{1-p} = {} & 1.738 - 1.332\,x_{B1\_1} - 0.185 x_{B2} + 0.209\,x_{B3\_1} + 1.201\,x_{B4\_1} - 1.038\,x_{B5\_1} \\ & - 0.538\,x_{B5\_2} - 0.577\,x_{H1\_1} + 0.520\,x_{H2\_1} + 0.379\,x_{H2\_2} - 2.223\,x_{H3\_1} + 1.053\,x_{H3\_2} \\ & + 1.025 x_{H3\_3} + 0.587 x_{H3\_4} + 0.326\,x_{E1\_1} + 0.289\,x_{E1\_2} + 0.140 x_{E1\_3} + 0.863\,x_{E1\_4} \\ & - 0.286\,x_{E2\_1} + 0.562\,x_{E2\_2} + 0.009\,x_{E2\_3} - 0.280\,x_{E2\_4} - 0.554\,x_{P1\_1} + 2.244\,x_{P1\_2} \\ & + 22.080 x_{P2\_1} - 0.159\,x_{P2\_2} + 0.384\,x_{P3\_1} - 0.032 x_{P3\_2} + 5.158\,x_{P4\_1} + 3.335\,x_{P4\_2} \\ & + 0.079 x_{C1\_1} + 0.654\,x_{C2\_1} \end{aligned}$$

$$\begin{aligned} M_D: \ln\frac{p}{1-p} = {} & 0.244 - 0.130 x_{B2} - 0.626 x_{P1\_1} + 1.767\,x_{P1\_2} + 19.616 x_{P2\_1} - 0.018\,x_{P2\_2} \\ & + 5.142\,x_{P4\_1} + 3.415\,x_{P4\_2} \end{aligned}$$

Nótese que las categorías de referencia no se incluyen puesto que su coeficiente es siempre cero. Puede observarse, por ejemplo, que hábitos de lectura más regulares y tendientes a temas profesionales o académicos se relacionan positivamente con una menor demora en el cursado. Interpretando el modelo reducido, los parámetros más significativos son la edad y la percepción de los egresados sobre la carrera. Los egresados con menor tiempo de cursado tienden a percibir la exigencia de la carrera como menor, y la extensión del plan de estudio como corta o adecuada.

Para la variable respuesta "Condición Actual", en el caso positivo dado por la categoría "Trabaja en relación a la profesión", el modelo saturado y el reducido están descritos por:

$$\begin{aligned} M_n: \ln\frac{p}{1-p} = {} & -7.254 - 3.619\,x_{B1\_1} + 0.291\,x_{B2} + 2.771 x_{B3\_1} - 3.902 x_{B4\_1} - 0.156\,x_{B5\_1} \\ & + 3.209\,x_{B5\_2} - 2.341 x_{H1\_1} + 0.612 x_{H2\_2} + 3.376 x_{H2\_3} + 4.930\,x_{H3\_1} + 1.381 x_{H3\_2} \\ & - 0.656 x_{H3\_4} - 0.837 x_{H3\_3} - 2.883 x_{E1\_1} + 0.007 x_{E1\_2} + 0.755 x_{E1\_3} + 0.249 x_{E1\_4} \end{aligned}$$

$$+0.411x_{E2\_1}+2.819x_{E2\_2}+1.252\,x_{E2\_3}+1.272x_{E2\_4}-3.176\,x_{P1\_1}-0.796\,x_{P1\_2}$$
$$+1.699x_{P2\_1}+2.218x_{P2\_2}+2.148\,x_{P3\_1}+0.118\,x_{P3\_2}-1.326\,x_{P4\_1}-0.847\,x_{P4\_2}$$
$$-2.724\,x_{M1\_1}-3.122x_{M1\_2}+1.642\,x_{M1\_3}-1.657\,x_{M1\_4}-1.205x_{M1\_5}+1.316\,x_{C1\_1}+2.479$$
$$x_{C1\_2}-2.272\,x_{C3\_1}-2.248x_{C3\_2}+1.713\,x_{C3\_3}-2.894\,x_{C3\_4}$$
$$-2.570\,x_{C3\_5}-3.234\,x_{C3\_6}$$

$$M_D: \ln\frac{p}{1-p} = -4.052+0.111x_{B2}-0.895x_{B1\_1}-1.305\,x_{B4\_1}+1.442\,x_{B5\_1}+2.359\,x_{B5\_2}$$
$$-1.053\,x_{H1\_1}+0.695x_{H2\_1}+1.599x_{H2\_2}-0.928x_{E1\_1}-0.552x_{E1\_2}$$
$$+0.805\,x_{E1\_3}+0.362x_{E1\_4}-1.634\,x_{P2\_1}+0.918x_{P2\_2}+0.786\,x_{C2\_1}$$

En este caso, en el método de reducción por criterio de la desviación no eliminó más variables que el método de reducción por AIC. En el modelo saturado, es interesante observar que la categoría "Desarrollo profesional" como motivo para estudiar en la universidad, así como "Flexibilidad y adaptación al cambio" como característica valorada por los empleadores son las únicas categorías en sus respectivas variables que se relacionan positivamente con la variable respuesta. En M_D, se observa una relación entre aquellos que trabajan en relación a la profesión y aquellos que tienen una edad mayor, viven solos o con familia propia, tienen una mayor cadencia en hábitos de lectura y mayores conocimientos de herramientas informáticas. En la Tabla 2 se muestra el desempeño de cada uno de los modelos obtenidos, incluyendo el modelo intermedio reducido sólo con el criterio AIC (M_{AIC}) y el modelo de redes neuronales (M_{NN}).

Tabla 2. Desempeño de los modelos para la variable "Tiempo de Cursado".

	Tiempo de Cursado				Condición Actual		
Métrica	M_n	M_{AIC}	M_D	M_{NN}	M_n	MAIC=MD	M_{NN}
PR	0.714	0.714	0.643	0.727	0.700	0.500	0.8125
RC	0.769	0.833	0.750	0.888	1.000	0.625	1.000
F1	0.741	0.769	0.692	0.799	0.824	0.556	0.896
MCC	0.542	0.610	0.477	0.537	0.783	0.382	0.450
AUC	0.786	0.807	0.786	0.831	0.776	0.662	0.750

4. Conclusiones

La metodología empleada, que combina el modelo Logit con métodos de reducción, permite obtener modelos comprensibles y significativos, que pueden brindar valiosa información en el contexto del análisis de datos de egresados para la toma de decisiones institucionales. Además, se implementó en R un método de reducción por eliminación hacia atrás basado en la desviación que será prontamente publicado en formato de librería de código abierto junto con funciones de evaluación de modelos.

References

1. Venables, W. N., Ripley B. D.: Modern Applied Statistics with S. 4ta edc, pp. 196-197. Springer, New York (2002).

2. Ripley, B., Venables, B., Bates, D., Hornik, K., Gebhardt, A., Firth, D: MASS: Support Functions and Datasets for Venables and Ripley's MASS, https://CRAN.R-project.org/package=MASS, last accessed 2023/08/01
3. Peña, D.: Análisis de Datos Multivariantes. 1ra edc. Mc Graw Hill Education, Madrid, (2002).
4. James, G., Witten, D., Hastie, T., Tibshirani, R. An Introduction to Statistical Learning with Applications in R, pp. 150-152. Springer, Nueva York (2013).

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Desarrollo de un chatbot y virtualización basada en contenedores para una aplicación web de gestión de laboratorios

María del Valle Scheffer[1] [0009-0002-9815-2346], Ilda Flavia Millán[1] [0009-0004-7457-3837], Fabio Gabriel Sanmartino[1] [0009-0003-9164-9533]

[1] Departamento de Informática, Facultad de Ciencias Exactas, Físicas y Naturales, Universidad Nacional de San Juan.

Complejo Islas Malvinas. Cereceto y Meglioli, 5400. Rivadavia. San Juan, 0264 4234129

maruscheff@gmail.com, flavia.millan@gmail.com, fabiosan221@gmail.com

Abstract. Este trabajo presenta el desarrollo de una tesis de grado, que tiene como objetivo la implementación de un Asistente Conversacional Virtual (ACV) o chatbot para la aplicación web del Laboratorio de Informática Aplicada (LIA) de la Facultad de Ciencias Exactas, Físicas y Naturales de la Universidad Nacional de San Juan, con el fin de mejorar la gestión de recursos y servicios en el LIA. El chatbot se utilizará como sistema de diálogo automatizado basado en Inteligencia Artificial (IA), que permitirá gestionar y enrutar las consultas de los usuarios de la aplicación web del LIA. Además, se propone la implementación de una infraestructura basada en contenedores de software para la virtualización de la aplicación web de gestión de recursos y servicios del LIA. Se espera que los resultados obtenidos, muestren que la implementación del chatbot y la virtualización basada en contenedores permitan una gestión más automatizada y eficiente de los recursos y servicios del LIA, con el fin de mejorar la eficiencia y efectividad en la gestión de recursos. En resumen, a partir de este trabajo se pretende optimizar la gestión de recursos y servicios en el Laboratorio de Informática Aplicada a través de un Asistente Conversacional Virtual y virtualización basada en contenedores, lo que representa una solución innovadora y eficiente para mejorar la atención y gestión de los recursos en el LIA.

Keywords: asistente virtual, chatbot, virtualización de aplicaciones web, contenedores de software, gestión de recursos y servicios en laboratorios informáticos.

1. Introducción

La gestión de recursos y servicios en el ámbito de la informática es un tema de gran importancia, ya que permite optimizar la atención y el uso de los recursos disponibles.

En este sentido, la implementación de tecnologías innovadoras y eficientes puede ser de gran ayuda para mejorar la calidad de la atención y la gestión de los recursos.

En este trabajo se presenta el desarrollo de una tesis de grado que tiene como objetivo la implementación de un ACV o chatbot para la aplicación web del Laboratorio de Informática Aplicada (LIA), con el fin de mejorar la gestión de recursos y servicios en el LIA. El chatbot se utilizará como sistema de diálogo automatizado basado en Inteligencia Artificial (IA), que permitirá gestionar y enrutar las consultas de los usuarios de la aplicación web del LIA. Además, se propone la implementación de una

infraestructura basada en contenedores de software para la virtualización de la aplicación web de gestión de recursos y servicios del LIA.

En relación a estas entidades vivientes artificiales con apariencia humana, conocidos como Asistentes Conversacionales Virtuales, chatbots o chatterbots, la tendencia actual de esta tecnología debe orientarse a desarrollar sistemas para dominios de aplicación restringidos, los cuales permiten utilizar información acerca de las posibles palabras, tipos de frases e intenciones de los usuarios [1]. De acuerdo a lo anterior, el chatbot propuesto en este trabajo se enfoca en la gestión de recursos y servicios en el LIA, lo que representa un dominio de aplicación restringido y específico.

En cuanto a la virtualización de la aplicación web del LIA, se propone la implementación de una infraestructura basada en contenedores de software, lo que permitiría una mayor flexibilidad y escalabilidad en la gestión de los recursos y servicios del LIA. En este sentido, la virtualización basada en contenedores de software permite una mayor eficiencia en la gestión de los recursos y servicios, ya que permite la creación de entornos de desarrollo estables y la simplificación del trabajo de mantenimiento de diferentes sistemas operativos [2].

En resumen, este trabajo se enfoca en la implementación de un ACV y virtualización basada en contenedores para mejorar la gestión de recursos y servicios que son ofrecidos en el Laboratorio de Informática Aplicada. Se espera que los resultados obtenidos permitan una gestión más automatizada y eficiente de los recursos y servicios del LIA, lo que reduciría costes en la atención de los usuarios y mejoraría los tiempos de respuesta a los usuarios y la gestión de los recursos del LIA, optimizando la calidad de la atención de los usuarios.

2. Contexto

El presente trabajo se enmarca en el ámbito de la gestión de laboratorios y la automatización de procesos. En particular, se enfoca en el Laboratorio de Informática Aplicada (LIA) dependiente del Departamento de Informática de la Facultad de Ciencias Exactas, Físicas y Naturales (FCEFN) de la Universidad Nacional de San Juan (UNSJ). El LIA, está conformado por cinco laboratorios cuyo principal objetivo es funcionar como espacios destinados a prácticas educativas universitarias de las carreras Licenciatura en Ciencias de la Computación, Licenciatura en Sistemas de Información y Tecnicatura Universitaria en Programación Web, dependientes del Departamento de Informática.

Por otro parte, el LIA brinda también servicios a través de su personal y de sus instalaciones para prácticas educativas que tienen como alcance al resto de los Departamentos que forman parte de la Facultad como así también de la Universidad Nacional de San Juan, incluso siendo sede en varias ocasiones de diversos encuentros y prácticas informáticas tanto a nivel provincial como nacional.

En el año 2002, se presentó la tesis cuyo título es “Proyecto de Reestructuración del Gabinete de Computación de la F.C.E.F. y N.”, que tenía como objetivo automatizar los procesos que se llevaban a cabo en el LIA a través de una aplicación de escritorio [3]. Esta propuesta, ha quedado desactualizada de acuerdo a las exigencias del siglo XXI

en cuanto a desarrollos de software y que brinde respuesta respecto a la gestión de recursos y servicios del LIA.

En el año 2020, se presentó un trabajo de tesis titulado "Aplicación web para gestionar la asignación de recursos y servicios de un Laboratorio de Informática Aplicada" [4]. En el mismo, se llevó a cabo el desarrollo de una aplicación web con el objetivo de brindar una mejora en la gestión de asignación de los recursos y/o servicios del LIA mediante la automatización de los procesos involucrados.

Bajo este contexto, el presente trabajo busca mejorar la eficiencia y la calidad de los procesos de gestión en el LIA a través de una aplicación web dinámica y un Asistente Conversacional Virtual basado en tecnologías de inteligencia artificial, específicamente un chatbot. El chatbot permitirá a los usuarios interactuar con la aplicación web de manera más natural y eficiente, brindando una experiencia de usuario mejorada. Además, se utilizará la tecnología de virtualización basada en contenedores para simplificar y dar utilidad real a la infraestructura de las Tecnologías de la Información del LIA, permitiendo una eficiente escalabilidad a la hora de redimensionar los servicios o aplicaciones, ahorrando el tiempo que se requiere cuando se realiza una nueva instalación.

3. Metodología

3.1 Diseño del Trabajo

La constante innovación tecnológica, hace que cada vez sea necesaria la aplicación y desarrollo de nuevos sistemas informáticos que se adapten a los nuevos tiempos, satisfaciendo las necesidades de sistemas y de usuarios.

La metodología que se adoptará para el presente trabajo, comenzará con una etapa exploratoria descriptiva y luego con una investigación documental. Se abordará el análisis y conocimiento de la aplicación web del LIA en la gestión de recursos y servicios. Luego, se iniciará la exploración, descripción e investigación de las diferentes metodologías de diseño de ACV o chatbots.

Seguidamente, se continuará con las tareas de exploración e investigación documental, para comparar y analizar las diferentes tecnologías existentes, en lo referido a la virtualización de sistemas web. Con lo anterior se propondrá finalmente la virtualización de la aplicación web de gestión de recursos y servicios, y se implementará la arquitectura en una infraestructura basada en contenedores de software.

Es importante destacar que, con la metodología de trabajo, se obtendrán conclusiones que refieren a las acciones de análisis, comparación y determinación de las tecnologías que se estudiarán e implementarán para dar cumplimento con el desarrollo del presente trabajo.

3.2 Definición de Variables

Se iniciará el estudio a partir de una lista de preguntas frecuentes discriminadas por categorías, correspondientes a las respuestas emitidas por docentes de la Facultad de Ciencias Exactas, Físicas y Naturales (FCEFN) de la Universidad Nacional de San Juan (UNSJ). Los datos corresponden a docentes y las variables corresponden a respuestas respecto a dudas y/o sugerencias con respecto a los recursos y servicios del LIA.

El análisis de los datos obtenidos permitirá obtener información que servirá para generar una base de datos a partir de la cual se tomen como entradas para el chatbot, y que le permita tomar decisiones para encauzar las respuestas de los usuarios de la aplicación web del LIA.

3.3 Instrumento de recolección de datos

La obtención de los datos se llevará a cabo a partir de una encuesta on line realizada a docentes de la Facultad de Ciencias Exactas, Físicas y Naturales de la Universidad Nacional de San Juan. El enfoque para el análisis e interpretación de los datos será del tipo cualitativo.

4. Conclusiones

En conclusión, el presente proyecto de tesis tiene como objetivo principal la implementación de un ACV o chatbot en la aplicación web del Laboratorio de Informática Aplicada (LIA) para orientar la gestión de recursos y servicios, y la virtualización de la aplicación mediante una arquitectura basada en contenedores de software.

La implementación del ACV y la virtualización de la aplicación web del LIA permitirá mejorar la eficiencia y efectividad en la gestión de recursos y servicios, así como una mayor flexibilidad y escalabilidad en la aplicación.

Es importante destacar que, a través de la metodología utilizada, se obtendrán conclusiones basadas en la investigación y experimentación, lo que permitirá identificar las mejores prácticas y tecnologías más adecuadas para la implementación del ACV o chatbot y la virtualización de la aplicación web del LIA.

4.1 Posibles líneas de trabajo futuro

A partir de la implementación del ACV o chatbot y la virtualización de la aplicación web del LIA, se abren diversas posibilidades de investigación futuras, que versarían sobre los siguientes temas, en referencia a:

1. Mejora en la precisión y eficacia del ACV o chatbot: Una posible línea de trabajo futuro sería, la mejora de la precisión y eficacia del ACV o chatbot, a través de la incorporación de técnicas de aprendizaje automático y procesamiento del lenguaje natural. Esto permitiría al sistema comprender mejor las consultas de los usuarios y proporcionar respuestas más precisas y útiles.
2. Integración con sistemas de gestión de incidencias: siendo otra posible línea de trabajo futuro, la integración del ACV o chatbot con sistemas de gestión de

incidencias, lo que permitiría automatizar la gestión de las solicitudes y consultas de los usuarios, y mejorar la eficiencia y efectividad en la atención a los mismos.

3. Ampliación de la base de conocimientos: De la cual se desprende como posible línea de trabajo futuro, la ampliación de la base de conocimientos del ACV o chatbot, a través de la incorporación de nuevas fuentes de información y la actualización constante de la misma. Esto permitiría al sistema proporcionar respuestas más precisas y actualizadas a las consultas de los usuarios.
4. Evaluación de la satisfacción de los usuarios: Finalmente, una línea de trabajo futuro importante sería, la evaluación de la satisfacción de los usuarios con el ACV o chatbot. Esto se llevaría a cabo, con la realización de encuestas y análisis de retroalimentación. Esto permitiría identificar posibles áreas de mejora y ajustar el sistema en consecuencia, para garantizar una experiencia satisfactoria para los usuarios.

Referencias

1. Morales-Rodríguez, M., Domínguez-Martínez, J.: Agentes conversacionales como un sistema de diálogo. Memorias del V Encuentro de Investigadores del ITCM, (2011).
2. Coronel, G. A., Vallejos, O.: Rendimiento de Aplicaciones Web en plataformas de contenedores de código abierto. In XX Workshop de Investigadores en Ciencias de la Computación (WICC 2018, Universidad Nacional del Nordeste), (2018).
3. Romagnano, M., Sánchez, E.: Reestructuración de una organización usando el proceso unificado del Rational: caso de estudio: reestructuración del gabinete de computación de la F.C.E.F.y N. dependiente de la U.N.S.J. Tesis de Grado (2002).
4. Sanmartino, F.: Aplicación web para gestionar la asignación de recursos y servicios de un Laboratorio de Informática Aplicada. Tesis de Grado (2020).

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Comparación de modelos de detección de objetos aplicados a una plantación de kiwi

Dante Mendoza, Camila Pagano, Silvia Pérez.

Escuela de Informática - Universidad Nacional del Oeste.
{hmendoza,cpagano,sperez}@uno.edu.ar

Resumen: Se propone evaluar la detección de órganos vegetales de kiwi a través del procesamiento automático de imágenes de la plantación. Se utilizan diferentes métodos de aprendizaje automático de la plataforma Tensorflow, evaluando su performance con indicadores de rendimiento y métricas de calidad sobre imágenes de prueba. Se utilizaron 11 familias de modelos, las que se evaluaron con métricas de calidad resultando solo una con alta calificación. Dicho modelo permitirá el conteo automático eficiente, adecuado a la realidad de campo.

Palabras claves: Detección de objetos en imágenes, Tensorflow, Colab.

1. Introducción

El conteo de órganos vegetales se realiza habitualmente para planificar tareas de raleo y prever la cosecha. Si bien el método clásico es el conteo en campo, este puede reemplazarse por conteo automático a partir del desarrollo de tecnologías de procesamiento de imágenes.

El aprendizaje automático tiene como objetivo permitir que un software realice una tarea sin la necesidad de programar reglas específicas. A partir de un conjunto de ejemplos, aprende patrones a partir de la información de entrada retenida, y puede continuar aprendiendo con nuevos ejemplos. La plataforma TensorFlow [1], de uso libre, ofrece un ecosistema completo de herramientas, bibliotecas y recursos, que permiten desarrollar proyectos relacionados con el aprendizaje automático .



Fig. 1: Imágenes de botones florales en una planta de kiwi.

En el marco de un proyecto de investigación se requiere realizar el conteo automático en imágenes de botones florales en una plantación de kiwi. El tipo de imágenes utilizadas puede verse en la Fig. 1, de un total de 1000 imágenes, las cuales

fueron separadas en conjuntos de entrenamiento, validación y prueba. Los dos primeros se utilizan para ajustar modelos pre-entrenados, de forma recursiva hasta lograr convergencia en parámetros de pérdida. Las imágenes de prueba permiten evaluar la calidad de cada modelo, comparando el resultado obtenido de la detección automática con el conteo visual realizado por un operador.

Aquí se presenta la comparación de modelos disponibles en Tensorflow, los que se evalúan a partir del tiempo de entrenamiento, consumo de memoria, y otros indicadores.

2. Metodología

La tecnología utilizada para la detección de objetos se basa en el uso de las redes neuronales convolucionales profundas (DCNN) [2]. A través de la API Tensorflow 2.8.1, se tomaron como base modelos pre-entrenados de modo de reducir el tiempo de ajuste del modelo final [3]. Para llevar a cabo la labor de etiquetado de objetos en las imágenes que los modelos utilizan para su etapa de entrenamiento, se utilizó el software libre LabelImage. Los modelos fueron probados con valores de configuración por defecto. Se consideraron diferentes modelos orientados a distintas aplicaciones como clasificación de imágenes, detección de objetos, conjunto de datos o integración con NodeJS.

Se realizó el entrenamiento usando COLAB [4] que ofrece 12.7 GB de RAM de sistema, 78.2 GB de Disco y una GPU, con un Tesla T4 con 15 GB.

La comparación de modelos se realizó clasificando su rendimiento según tiempo de entrenamiento, consumo de memoria RAM y GPU, y resultados en la detección. Si en todos estos aspectos el rendimiento es malo, se considera que el modelo es Malo ó Muy malo. Si los tiempos de entrenamiento son relativamente cortos (por debajo de 10 minutos para 100 pasos de entrenamiento), el consumo de memoria está entre 4 a 16 GB de RAM, y la detección es aceptable, se consideró al modelo Bueno/Regular. Finalmente, si todos los indicadores dan aceptables, el modelo se considera Muy Bueno.

En imágenes de prueba se evaluó también la cantidad de objetos efectivamente detectados (VP, verdaderos positivos), la cantidad de objetos presentes en la imagen y no detectados (FN, falsos negativos) y la cantidad de detecciones que no corresponden a objetos realmente (FP, falsos positivos). En función a estos, se tienen las siguientes métricas usuales de calidad:

Sensibilidad: proporción de casos positivos identificados correctamente, con respecto a todos los casos positivos reales. Se calcula como Sensibilidad = VP / (VP + FN) **Precisión:** proporción de casos positivos identificados correctamente, con respecto a todos identificados como positivos. Se calcula como Precisión= VP / (VP + FP) Por otra parte, se tomaron en cuenta otras características para cada modelo preentrenado: la velocidad (speed), medida en milisegundos, y Mean Average Precision(mAP), que es una métrica utilizada para evaluar el rendimiento de modelos de detección de objetos. Dichos valores se encuentran publicados en el sitio oficial de Tensorflow. También se tomó el tiempo promedio, en minutos, que tardó cada modelo en completar cien pasos de entrenamiento.

3. Resultados obtenidos

Se consideraron diferentes familias de modelos, entrenando con las imágenes de botones florales hasta cinco mil pasos.

Para la familia de modelos *CenterNet HourGlass104*, se observó que en doscientos pasos de entrenamiento el proceso colapsaba por la alta demanda de memoria de GPU, llegando al límite de memoria disponible. El modelo no pudo entonces probarse, dado que requiere de un poder de cómputo superior al disponible. Lo mismo ocurrió con la familia *CenterNet MobileNetV2*, el cual demandaba un alto consumo en memoria RAM y GPU. En general, la familia *CenterNet Resnet* mostró una alta presencia de falsos negativos en todas sus variantes.

A su vez, la familia *EfficientDet* en sus variantes del D0 a D3, mostraron presencia alta de falsos negativos, con alto tiempo para cien pasos de entrenamiento, lo cual lo hace un modelo de alta demanda de memoria RAM y GPU. En las variantes D4 a D7 los cien pasos de entrenamiento no concluyen luego de cuatro horas, por lo que se decidió abortar su ejecución. Se considera que requiere de un hardware de altas prestaciones.

La familia *SSD MobileNet* no logró entrenar cien pasos de entrenamiento en dos horas, pero su variante V2 FPNLite 640x640 pudo concluir su entrenamiento, aunque no logró detecciones en las imágenes de prueba. Esto lleva a considerar estos modelos como de muy mal rendimiento.

En el caso de la familia *SSD ResNet* se tienen resultados mixtos, ya que su variante V1 FPN 640x640 no logra hacer ninguna detección, pero en el resto de variantes logra hacer detecciones, aunque con un porcentaje muy bajo de verdaderos positivos. Resultados similares se obtuvieron con el modelo *Mask R-CNN Inception*: el tiempo requerido para cien pasos de entrenamiento fue de una hora, con unas detecciones muy poco satisfactorias.

La familia *Faster R-CNN ResNet* obtuvo resultados más que aceptables en las detecciones y con un promedio de diez minutos por cada cien pasos de entrenamiento. Si bien se observaron falsos negativos, no fueron significativos lo que permite inferir que. con más pasos de entrenamiento y haciendo ajustes en su configuración por defecto, se puede mejorar estos resultados obtenidos.

Finalmente, la familia *Faster R-CNN Inception ResNet V2* obtuvo detecciones muy satisfactorias, con un tiempo de trece minutos para cien pasos de entrenamiento y poca presencia de falsos negativos. La demanda de memoria RAM y GPU fue media- alta, en el caso de la RAM utilizaba unos 10GB del total disponible, mientras que la GPU demandaba unos 12GB de memoria.

A partir de este análisis resulta que los modelos de la familia *Faster R-CNN Inception ResNet V2* fueron los que más se destacaron en los resultados obtenidos, manteniendo un equilibrio entre demanda de recursos, tiempos de entrenamiento y objetivos alcanzados. En la Tabla 1 se puede apreciar una comparación en las detecciones que obtuvo el mencionado modelo contra otro de los analizados, de diferente familia. Se observa aquí gran diferencia en las detecciones correctas (VP_Faster = 115 versus VP_Center = 45), tanto como en los objetos no detectados (FN_Faster = 21 vs FN_Center = 91).

Tabla 1. Comparación en imágenes de prueba de dos modelos analizados.

	Faster_rcnn_inception_resnet			CenterNet_resnet101_v1_fpn		
Imagen	**VP**	**FP**	**FN**	**VP**	**FP**	**FN**
Imagen 1	5	0	6	1	0	10
Imagen 2	6	2	5	5	0	6
Imagen 3	25	0	2	5	0	22
Imagen 4	8	0	0	4	0	4
Imagen 5	19	0	2	3	0	18
Imagen 6	9	0	2	3	0	8
Imagen 7	13	0	1	10	0	4
Imagen 8	6	0	0	3	0	3
Imagen 9	8	0	2	5	0	5
Imagen 10	16	2	1	6	0	11
TOTAL	115	4	21	45	0	91

Por otro lado, en la tabla 2 se puede apreciar un resumen de las familias de modelos analizados. Se convino que aquellos modelos que no pudieron concluir su etapa de entrenamiento fueron considerados como no probados ya que se requiere de un hardware más potente para obtener unos resultados aceptables [5].

Tabla 2. Resumen de las familias de modelos analizados.

Modelo	**Speed (ms)**	**mAP**	**Tiempo aprox. cada 100 pasos**	**Resultado**
CenterNet HourGlass104	70	41.9	Sin concluir	No probado
CenterNet Resnet	30	27.6	0.5 minutos	Malo
CenterNet MobileNet	6	23.4	Sin concluir	No probado
EfficientDet D0 - D3	95	45.4	30 minutos	Malo
EfficientDet D4 - D7	325	51.2	Sin concluir	No probado
SSD MobileNet V1	22	22.2	Sin concluir	No probado
SSD MobileNet V2	39	28.2	1 minuto	Muy malo
SSD ResNet	111	39.6	8 minutos	Muy malo
Faster R-CNN ResNet	101	37.4	10 minutos	Regular/Bueno
Faster R-CNN Inception	236	37.7	13 minutos	Muy bueno
Mask R-CNN Inception ResNet V2	301	39.0	60 minutos	Malo

4. Conclusiones

La comparación entre diferentes modelos de detección de objetos en imágenes es de interés para lograr automatizar el proceso de conteo de órganos frutales en una plantación [6].

En este trabajo se ajustaron algunos modelos de uso extendido en la detección de objetos, evaluándose sobre un reducido conjunto de pruebas. Los resultados fueron muy variables: en algunos casos no se obtuvieron detecciones, en otros no se logró completar en tiempos razonables la etapa de entrenamiento. Así también, en algunos modelos se consiguió detectar objetos, aunque con un bajo rendimiento.

La familia *Faster R-CNN Inception ResNet V2* fue la que mejores resultados obtuvo, con tiempos de entrenamiento aceptables y buen resultado en las detecciones de botones florales. En cuanto a esto último, se evaluó precisión y sensibilidad en las 10 imágenes de prueba, las que dieron valores 0,84 y 0,96 muy por encima de otros modelos.

En futuros trabajos se propone evaluar estos modelos en otras etapas fenológicas de los kiwis, como así también utilizar técnicas de aumento de datos artificial, observando su utilidad en función al aumento de tiempos de entrenamiento y una demanda de memoria RAM y GPU.

Referencias

[1] TensorFlow Homepage, https://www.tensorflow.org/?hl=es-419, accedido el 27 de julio del 2023.

[2] Pérez, S. N., Dejean, G. A., Giuliano, M., Yommi, A., David, M. A., Murillo, N. L., Mendoza, D. H.: Detección de yemas brotadas para la estimación temprana del rendimiento de una plantación de kiwi. In: XXIV Workshop de Investigadores en Ciencias de la Computación, WICC 2022, pp. 200-204. Mendoza, Argentina (2022).

[3] García Santos, K.: Sistema de Detección de Objetos para Reconocimiento Gestual mediante Redes Neuronales Convolucionales, Tesis. Universidad Politécnica de Valencia (2019).

[4] Bisong, E.: Google colaboratory. In: Building machine learning and deep learning models on google cloud platform: a comprehensive guide for beginners. 1st ed, pp. 59-64. Apress, Berkeley, CA (2019).

[5] Mendoza, D.: Estimación temprana de rendimiento del cultivo de kiwi mediante el procesamiento de imágenes. Tesis de Licenciatura en Informática, Universidad Nacional del Oeste (2021).

[6] Dejean, G. A., Pérez, S. N., Mendoza, D. H., Giuliano, M.: Detección de objetos aplicada al cultivo de kiwi. En: CONAIISI 2022, Décimo Congreso Nacional de Ingeniería Informática y Sistemas de Información, pp 158-164. Entre Ríos, Argentina (2022).

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

EcoConversa: Contribuyendo al futuro sostenible con Inteligencia Artificial

Franco Brandan[1] [0009-0004-9790-3501]; Andrea Villagra[1] [0000-0003-1285-2959] y Daniel Pandolfi[1] [0009-0000-5403-8993]

[1] Universidad Nacional de la Patagonia Austral, Unidad Académica Caleta Olivia, Argentina
Instituto de Tecnologías Aplicadas
{fbrandan, avillagra, dpandolfi}@uaco.unpa.edu.ar

Resumen. El presente trabajo se enfoca en el desarrollo de "EcoConversa", un chatbot basado en Inteligencia Artificial entrenado para abordar los Objetivos de Desarrollo Sostenible (ODS) de la ONU planteados para el 2030, con especial atención en los ODS 11 y 17. EcoConversa utiliza técnicas avanzadas de Procesamiento del Lenguaje Natural (NLP) y aprendizaje automático para interpretar las intenciones de los usuarios y brindar respuestas adecuadas.

El chatbot fue construido utilizando el *framework* Rasa y se entrenó con una amplia variedad de datos, incluyendo artículos de Wikipedia y un corpus lingüístico especializado para mejorar la comprensión del lenguaje informal. Los resultados demuestran una identificación correcta del 84% de las intenciones de los usuarios. EcoConversa se presenta como una herramienta prometedora para la difusión y concientización sobre el desarrollo sostenible.

Palabras clave: Procesamiento de Lenguaje Natural, Inteligencia Artificial, Objetivos de Desarrollo Sostenible.

1. Introducción

En el mundo moderno caracterizado por su interconexión global y los avances tecnológicos vertiginosos es imperativo tener en cuenta los efectos que tendrán a futuro la implementación de estas herramientas en la vida cotidiana de las personas [2, 3]. El mayor desafío al que nos enfrentamos como sociedad es el garantizar un futuro sostenible para las siguientes generaciones, y para esto Naciones Unidas plantea 17 Objetivos de Desarrollo Sostenible para el año 2030. Lograr estos objetivos no es una tarea sencilla. Se requiere una combinación de esfuerzos, cooperación y la aplicación efectiva de herramientas capaces de amplificar el impacto de nuestras acciones.

A partir de este contexto surge el desarrollo de un asistente virtual, denominado “EcoConversa”, basado en Inteligencia Artificial (IA) y entrenado para mantener conversaciones informativas y educativas sobre esta temática crucial. En específico se enfoca en los ODS 11 (Ciudades y Comunidades Sostenibles) y 17 (Alianzas para lograr los objetivos), esenciales para abordar el desarrollo urbano sostenible y la cooperación global.

Los sistemas inteligentes artificiales como este asistente son herramientas adaptables que permiten a los usuarios interactuar de manera directa y natural [8]. Estos son capaces de comprender la información recibida de manera que se puede predecir una salida y responder acorde a la situación actual basándose en el modelo de aprendizaje. Se evalúa la confianza con que se identifica la intención real del usuario y

luego es elegida la salida con el mayor puntaje de acierto [9]. De esta manera simulan una conversación humana, facilitando el acceso a información específica sin requerir la interacción directa con una persona. Pueden abarcar desde pequeños agentes conversacionales hasta asistentes virtuales complejos capaces de relacionar múltiples conceptos y conversaciones, logrando un gran nivel de caracterización propia e "inteligencia" [5, 11].

El presente artículo está organizado de la siguiente manera: la Sección 2 exhibe la tecnología y metodologías utilizadas para la construcción y entrenamiento del modelo de lenguaje y posteriormente se realiza un análisis de los resultados obtenidos. Finalmente, la Sección 3 presenta las conclusiones obtenidas del proyecto desarrollado.

2. Desarrollo

2.1 Construcción y entrenamiento

El chatbot EcoConversa fue creado utilizando el *framework* de Rasa[1] basado en Python y de código abierto. Se eligió esta herramienta por su arquitectura modular, facilitando a futuro la modificación y creación de nuevas funcionalidades [6]. Consta de una IA que puede procesar entradas y entregar mensajes de acuerdo con el contexto de la conversación que se está manteniendo. Se utiliza esta tecnología y la aplicación de algoritmos de aprendizaje automático (Machine Learning) para el NLP [4]. Un ejemplo de agente conversacional similar es ChatGPT [1], que consta de una super IA con un largo proceso de entrenamiento utilizando aprendizaje por refuerzo a partir de retroalimentación humana (RLHF) y transformador que permite el análisis y comprensión de los textos humanos en lenguaje natural en grandes cantidades. Este método de entrenamiento característico de GPT provee una funcionalidad mejor alineada a las intenciones humanas.

A partir de la base provista por Rasa en inglés, se comienza el proceso de mejora y expansión del modelo de lenguaje para el aprendizaje del lenguaje español. Con este objetivo se hace entrega de una colección de artículos de Wikipedia para familiarizar a la IA con la sintaxis del idioma [2].

Al utilizar los artículos descritos en el párrafo anterior, se obtuvo una correcta lematización formal del lenguaje, aunque se observó una falta importante de precisión cuando el usuario se comunica de manera informal. Para solucionar esto se utilizó AnCora-ES3[2] [10], un corpus lingüístico extenso constituido principalmente por textos periodísticos curados para la diversidad recopilando 500.000 palabras distintas y clasificadas por lemas y categorías morfológicas, funciones sintácticas, clase semántica verbal, entre otras clasificaciones. En la Tabla 1 se puede observar un ejemplo de la clasificación por anotación morfológica, asociando un lema, la categoría y atributos morfológicos de cada palabra en el corpus. Luego, se utilizó un *dataset* [7] que recopila 7.200 artículos analizados manualmente para la clasificación de entidades, para lograr el reconocimiento de nombres propios en una primera instancia [3]. En la siguiente

[1] Página oficial de Rasa: https://rasa.com
[2] Corpus AnCora-ES: http://clic.ub.edu/corpus/es

Figura 1 se puede observar los aportes provistos por cada *dataset* al modelo de lenguaje. La primera colección de artículos que brindaron la comprensión de la sintaxis del español, la inclusión de AnCora permitió la comprensión de lenguaje informal, y por último al implementar el *dataset* de artículos clasificados se obtuvo una mejora en la identificación y clasificación de entidades.

Tabla 1. Salida del análisis morfológico de la oración "Si trabajo bajo presión bajo la atención." Adaptado de [10]

Palabra	Lema1	PoS1	Lema2	PoS2	Lema3	PoS3	Lema4	PoS4
Si	si	CS	si	NCMS	si	RG	-	
trabajo	trabajar	VMIP1S	trabajo	NCMS	-			
bajo	bajar	VMIP1S	bajo	AQ0MS	bajo	CMS	bajo	SPS
presión	presión	NCFS	-					
bajo	bajar	VMIP1S	bajo	AQ0MS	bajo	NCMS	bajo	SPS
la	la	DA0FS	-					
atención	atención	**NCFS**	-					
.	.	Fp	-					

Cada dígito expresa una pieza de información. Analizaremos el PoS (etiquetado gramatical) de atención (NCFS).
El primer dígito expresa la categoría principal (**N**=Noun/**Sustantivo**), luego la subcategoría (**C**=Common/**Común** o P=Proper/Propio), el tercer dígito es el género (M=Masculino, **F**=**Femenino**, C=No especificado). Finalmente, la cuarta posición muestra el número (**S**=**Singular**, P=Plural).

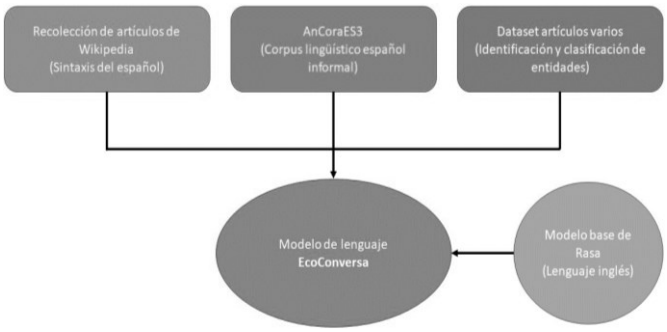


Fig. 1. Contribución de cada *dataset* al modelo de lenguaje

2.2 Resultados y análisis

Para la evaluación de los resultados obtenidos del entrenamiento de este modelo se realizó un análisis de confianza de las entradas recibidas por el chatbot y se compararon con las intenciones reales de los usuarios. En total se contempló un total de 19 caminos distintos de conversación general sobre los ODS y la ONU; una descripción breve para cada objetivo (17); 8 caminos en específico sobre el ODS 11 y 7 caminos sobre el ODS 17; formando un total de 51 caminos únicos de información que el agente conversacional debe identificar correctamente.

Los resultados obtenidos se recopilaron en forma de porcentajes indicando la tasa de acierto en la identificación de intención llegando a un total de 84% de certeza al

analizar una entrada. Esto significa que, de 100 peticiones realizadas al modelo, 84 son identificadas correctamente al primer intento. Las entradas identificadas incorrectamente luego son corregidas a través de un entrenamiento manual. Durante la ejecución, el chatbot utiliza un mensaje *fallback* al encontrarse en una situación con muy baja confianza (<40% de confianza) solicitando al usuario reformule su mensaje para una mejor comprensión.

En la Tabla 2 pueden observarse algunos ejemplos de entradas de difícil comprensión analizadas junto a su intención real y la intención identificada por el modelo. Por último, se encuentra el valor de confianza (también representado en porcentaje) del modelo al identificar la intención y el total obtenido en general.

Tabla 2. Resultados de análisis de confianza

Entrada	Intención real	Intención identificada	Valor de confianza
Hola kmo te va?	Greet		0,633 (63%)
Q tl?	Greet		0,964 (96%)
Hola, Ferchu, qué onda?	Saludo		0,999 (99%)
Adioh!	Despedida		0,999 (99%)
Chauu, cuidteh.	Despedida		0,780 (78%)
Nus vemoh!	Despedida		0,996 (99%)
Hi, how are you doing?	Greet (inglés)	NLU_Fallback	0,5 (50%)
Cmo hago para ablr con la ofcna	Contacto		0,971 (97%)
Cual ees el numro?	Contacto		0,680 (68%)
...			
Porcentaje de entradas identificadas correctamente			**0,84 (84%)**

Puede observarse un buen nivel de aciertos de un lenguaje informal según su contexto a pesar de los posibles errores gramaticales. El tamaño extenso de los datos de entrenamiento mejoró la precisión del chatbot en la identificación de intenciones.

3. Conclusión

Durante el desarrollo de este proyecto se utilizaron técnicas y recursos del estado del arte basadas en español para la creación de un modelo que sienta las bases para futuras investigaciones y desarrollos en el campo de la inteligencia artificial. Este enfoque evolutivo continuo es importante para mantener su relevancia y eficacia en un entorno volátil con cambios constantes.

A pesar del gran desafío que presenta la comprensión del lenguaje natural los resultados obtenidos hasta el momento (84% de confianza) demuestran una significativa precisión en la identificación de las intenciones reales de los usuarios, convirtiendo a "EcoConversa" en una herramienta prometedora para la difusión y concientización sobre el desarrollo sostenible.

A futuro se planea explorar la posibilidad de integrar técnicas avanzadas de aprendizaje automático, como el aprendizaje por refuerzo adoptado por ChatGPT, para mejorar la capacidad de adaptación del modelo a nuevas situaciones y escenarios no contemplados en el entrenamiento inicial. Además de permitir a personas ajenas al ámbito interactuar libremente con el agente conversacional con el objetivo de recopilar nuevos datos de entrenamiento sobre el uso real de la aplicación.

Referencias

1. Bang, Y., Cahyawijaya, S., Lee, N., et. al.: A Multitask, Multilingual, Multimodal Evaluation of ChatGPT on Reasoning, Hallucination, and Interactivity. arXiv:2302.04023v2. (2023)
2. Brandan, F., Pandolfi, D., Villagra, A.: Sistemas conversacionales aplicados a la gobernanza Asistencia automatizada al público. Informes Científicos Técnicos – UNPA, 14(3), 44-68. https://doi.org/10.22305/ict-unpa.v14.n3.894 (2022)
3. Brandan, F., Villagra, A., Pandolfi, D.: Sistema inteligente conversacional aplicado a la gobernanza local: Automatización de asistencia pública. Informes Científicos Técnicos – UNPA, 15(3), 49-69. https://doi.org/10.22305/ict-unpa.v15.n3.983 (2023)
4. Bunk, T., Varshneva, D., Vlasov, V., Nichol, A.: Diet: lightweight Language Understanding for dialogue systems. https://arxiv.org/pdf/2004.09936.pdf (2020)
5. Hussain, Z., Hassan, A.: Chatbots and their ethical challenges. Journal of Information, Communication and Ethics in Society, 18(2), 192-209. (2020)
6. Jiao, A.: An intelligent chatbot system based on entity extraction using Rasa NLU and Neural Network. Journal of Physics: Conference Series, 1487(1). (2020)
7. Nothman, J., Ringland, N., Radford, W., Murphy, T., Curran, J. R.: Learning multilingual named entity recognition from Wikipedia. Artificial Intelligence, 194, 151–175. (2013)
8. Oracle What is a Chatbot?, https://www.oracle.com/chatbots/what-is-a-chatbot/, last accessed 2023/07/31
9. Pengfei, L., Weizhe, Y, Jinlan, F, et. al.: Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing arXiv:2107.13586v1. (2021)
10. Taulé, M., Martí, M.A., Recasens, M.: Ancora: Multilevel Annotated Corpora for Catalan and Spanish, Proc. of 6th Int. Conf. on Language Resources and Evaluation. (2008)
11. Wayne, X., Kun, Z., Junyi, L., et. al.: A survey of Large Language Models. arXiv:2303.18223v8. (2023)

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Sistema de control y monitoreo para Biorreactor con tecnología IoT

Mauro Escalante[1], Diego Irusta[1] Gustavo Mercado[2], Carlos Taffernaberry[2], Ana Laura Diedrichs[2], Sebastian Tobar[2], Cristian Pérez Monte[2], Eugenio Fisicaro[3], Adrian Carmona[3], Ana Lattuca[4], Nelson Merino[4], Néstor Manzur[4], Antonio Álvarez[4].

[1] Ingeniería Electrónica, Facultad Regional Mendoza, UTN
maurom.i.e12@gmail.com, diegomartinirusta@gmail.com

[2] gridTICs, Ingeniería Electrónica, Facultad Regional Mendoza, UTN
{gmercado, cartaffe, ana.diedrichs, mstobar,cfperez}@frm.utn.edu.ar

[3] Planta Piloto, Ingeniería Química, Facultad Regional Mendoza, UTN
carmonaadrian34Q@gmail.com, ingenio@ingeniorecursos.com.ar

[4] Cátedra Proyecto Final, Ingeniería Electrónica, Facultad Regional Mendoza, UTN
ana.lattuca@docentes.frm.utn.edu.ar, nelson.merino@frm.utn.edu.ar,
nestor.manzur@gmail.com, antonioalvarezabril@yahoo.com.ar

Abstract. El proyecto propone la implementación de un sistema de control y monitoreo para un biorreactor de mezclado en la planta piloto de Química de la UTN Facultad Regional Mendoza. Mediante un sistema de control, se mantendrán condiciones ambientales óptimas para que los agentes biológicos puedan sobrevivir y reproducirse. Se implementará la tecnología IoT y motes, como así también un monitoreo local y remoto, lo cual permitirá al usuario supervisar la reacción biológica y tomar acciones necesarias a través de la observación de gráficos, estadísticas, históricos, avisos y alarmas. El proyecto busca facilitar investigaciones científicas en la universidad, así como ser objeto de estudio para alumnos y docentes de diferentes áreas relacionadas. Además, se destaca la relevancia del uso de biorreactores en la industria, con potencial en el mercado. El formato del proyecto puede ser escalado a uno o más biorreactores de mayor tamaño, como los utilizados en la industria alimentaria, bebidas, química entre otras. Se realizará una transferencia de tecnología, proporcionando información del manejo, diseño, reconstrucción, reparación y ampliación del proyecto. Cabe destacar que el proyecto cuenta con el financiamiento del departamento de Química. Y soporte técnico por parte del personal de la planta piloto de Química y del grupo gridTICs.

Keywords: IoT, biorreactor, control, monitoreo, motes, transferencia de tecnología,

Contexto

El presente trabajo está inserto en el Proyecto Final de Grado denominado “SISTEMA DE CONTROL Y MONITOREO PARA BIORREACTOR CON IOT”, acreditado por la Cátedra Proyecto Final de la Carrera de Ingeniería de Electrónica de la Universidad Tecnológica Nacional y que se desarrolla en durante el año lectivo de 2023.

El trabajo final es llevado adelante por alumnos tesistas y con la supervisión de docentes de la cátedra. Mientras que el grupo GridTICs tiene las tareas de dirección

técnica, asesoramiento y provisión de laboratorios. Por otro lado, la Planta Piloto actúa como stakeholder y entidad de especificación de control bioquímico del biorreactor. Adicionalmente. la Planta Piloto, recibe el producido por este proyecto, y verifica su cumpliento con las especificaciones.



Figura 1 Sistema Biorreactor

1. Introducción

En la planta piloto de ingeniería química, de nuestra facultad, existe la necesidad de poner en funcionamiento un biorreactor [1] de mezclado para estudiar diversas reacciones biológicas. Además, dicho biorreactor debe poder repararse, reproducirse y ampliarse en un futuro. ver Figura 1

Para ello implementaremos un sistema eficiente de control y monitoreo. basado en tecnología de IoT (Internet of Things - Internet de las Cosas) [2], posibilitando sus aplicaciones en la investigación, la educación y la industria. También se realizará una transferencia de tecnología para garantizar la escalabilidad y manejo del sistema. En la figura 2 se puede observar el diagrama en esquemático del sistema donde se muestran los componentes mecánicos, electrónicos, de comunicaciones e informáticos que forman el sistema.

1.1 Justificación del proyecto (Caso de Negocio)

1.1.1 Antecedentes del proyecto

Este proyecto nace como iniciativa del Departamento de Química de nuestra Facultad con el objetivo de ampliar el estudio de reacciones biológicas. Sin embargo, con el paso de los años, el proyecto quedó obsoleto. Por tanto, vamos a reactivar esta iniciativa, ya que tiene un gran potencial para el futuro de la facultad y la posibilidad de escalar el proyecto en la industria. Además, consideramos que esto incentivará la aparición de proyectos interdisciplinarios cada vez más frecuentes en nuestra Facultad. Finalmente, creemos que la implementación de nuestro sistema de control y monitoreo resultará en un biorreactor totalmente funcional para su uso en el Departamento de Química.

1.1.2 Estado actual

Actualmente se cuenta con el tanque del biorreactor de mezclado y toda la estructura necesaria, un variador de frecuencia, un compresor y un motor de inducción, así como el espacio físico para llevar a cabo la instalación. Cabe mencionar que el biorreactor no está en funcionamiento y hasta la fecha nunca lo ha estado.

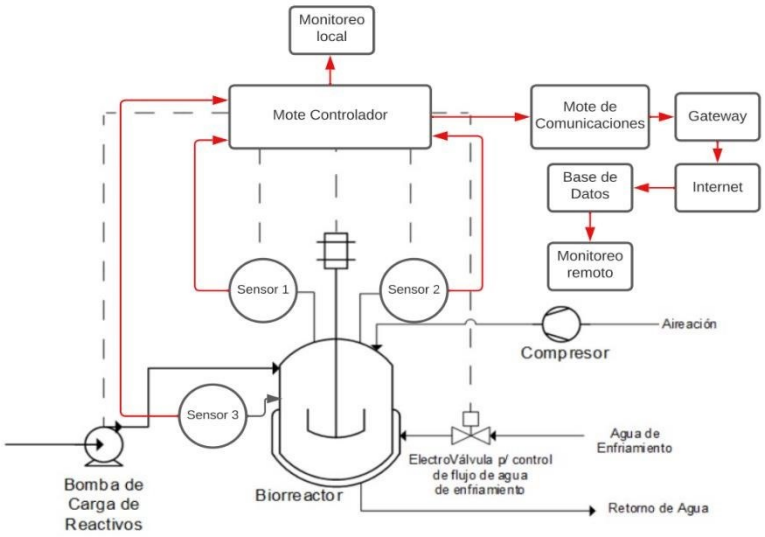


Fig. 2. Diagrama esquemático funcional del Sistema de control y monitoreo para Biorreactor

En cuanto al campo de aplicación, aunque nuestro proyecto no se implementará a nivel industrial, tiene el potencial de ser escalable a la industria con los cuidados y ajustes necesarios. En la actualidad, existen biorreactores en la industria que están conectados a sistemas SCADA o cuentan con interfaces HMI. Si bien hay biorreactores con conexión a internet y monitoreo remoto, no parecen ser muy comunes en este tipo de dispositivos. No se han encontrado registros del uso de IoT y motes en biorreactores, aunque no se descarta su existencia

1.1.3 Necesidad del negocio y definición del problema

Las necesidades a cubrir son las siguientes:

- Contar con una herramienta que permita el control inteligente del cultivo de agentes biológicos en la planta piloto del Departamento de Química de nuestra facultad.
- Implementar un sistema de monitoreo remoto con alarmas para gestionar las reacciones aleatorias y ahorrar tiempo al personal encargado.
- Establecer un sistema de monitoreo local.
- Mejorar la calidad de la educación en la facultad.
- Fortalecer el área de investigación de la facultad.

- Mejorar las instalaciones de la facultad.
- Contar con un medio para realizar el mantenimiento, reparación, reproducción y mejoras del biorreactor.

Específicamente, el problema que buscamos solucionar es la falta de un biorreactor funcional en la planta piloto del Departamento de Química de nuestra Facultad. Aunque contamos con la estructura del biorreactor, este no está en funcionamiento debido a la falta de instalación y de los sistemas de control, monitoreo y alarmas necesarios. Además, es necesario contar con un medio que permita llevar a cabo el mantenimiento, reparación, reproducción y mejoras del biorreactor.

Nuestro proyecto tiene como objetivo abordar este problema mediante la implementación de un sistema de control, monitoreo y alarmas, así como mediante la transferencia de tecnología necesaria para su funcionamiento.

2. Objetivo

Se busca realizar un sistema de control y monitoreo para las variables críticas de un Biorreactor. Las variables críticas principales que se buscan controlar y cambiar en el proceso son: la temperatura y el pH.Para ello se implementará un sistema de control implementado por microcontrolador. También se medirá oxígeno disuelto o potencial redox. Además para poder realizar el monitoreo remoto de dichas variables, a través de tecnologías de IoT.

Se realizará una transferencia de tecnología a través de un plan de documentación.

3. Descripción

Como se muestra en la figura 3 existen diferentes módulos que componen el sistema de control y monitoreo del biorreactor.

El sistema de control se basa en monitorear las variables críticas principales - temperatura y el pH- y actuar para mantener el rango programado mediante actuadores. Los actuadores serán una electroválvula para hacer circular el agua por el sistema intercambiador de calor, una bomba de reactivos para el pH y un compresor para mantener aireada la reacción con oxígeno. El sistema de control podría ser implementado mediante control PID e instalado en el denominado Mote 1 Controlador (ver fig 3). El Mote 1 es un sistema embebido compuesto por microcontrolador, memorias, sistemas de I/O y comunicaciones seriales locales. Este sistema embebido probablemente sea implementado por sistema Arduino Industrial [3] [4] y/o ESP 32 [5]. Este Mote tiene asociado el monitoreo local que se realizará a través de una pantalla LCD junto con un teclado. En pantalla LCD se mostrarán los valores de los parámetros sensados y el teclado funcionará para navegar por la pantalla y establecer los SetPoint de control

El monitoreo remoto se llevará a cabo mediante técnicas de IoT, es decir:

Sensor-Mote-Comunicaciones-Almacenamiento-Visualización-Análisis

Como se mencionó en el párrafo anterior, el esquema **sensor-mote** es realizado por el Mote 1 y se agrega un segundo dispositivo, denominado Mote 2 para implementar la etapa de comunicaciones. En este mote se instala un stack TCP/IP que permite implementar la comunicación IoT. Para la capa física y de enlace, se elige el estándar IEEE 802.11[6], aprovechando la red wifi disponible en las instalaciones de la Planta Piloto. Para las demás capas se prefieren los protocolos estándares recomendados por el IETF [7] para las tecnologías de IoT, tales como: 6lowpan (capa de Internet), udp (capa de transporte) y CoAP (Capa de aplicación).

La etapa de almacenamiento se implementa por un sistema de base de datos, denominado gridTICs-cloud y la etapa de visualización y análisis de implementa a través de un sistema dashboard [8] (a ser definido). Conformando, estas dos últimas etapas, un sistema backend/frontend. Por lo tanto los usuarios remotos podrán disponer de la información de monitoreo del biorreactor a través de la web.

Cabe destacar, que mediante el sistema de visualización también será posible modificar los puntos de ajuste del control del biorreactor de forma remota.

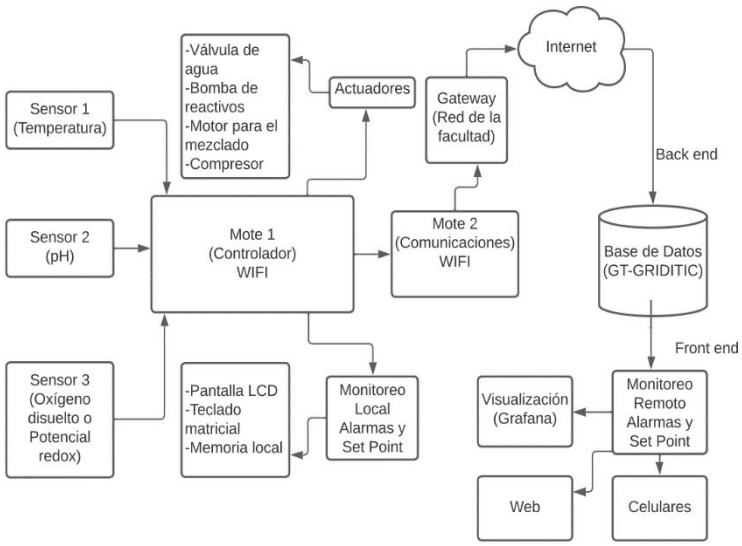


Fig. 3. Diagrama en bloques general del Sistema de control y monitoreo para Biorreactor

Referencias

1. Biorreactor, https://www.cytivalifesciences.com/en/us/news-center/what-is-a-bioreactor10001, last accessed 2023/06/20
2. IoT, https://www.oracle.com/ar/internet-of-things/what-is-iot/, last accessed 2023/06/25
3. ARSHEEN MIR, R. SWARNALATHA, "IMPLEMENTATION OF AN INDUSTRIAL AUTOMATION SYSTEM MODEL USING AN ARDUINO" Journal of Engineering
4. Science and Technology Vol. 13, No. 12 (2018) 4131 - 4144
5. Arduino Industrial - Industrial Shields https://www.industrialshields.com/es_ES/industrial-automation-solutions-home-20220310-lp, last accessed 2023/06/10

6. ESP32 Espressif, https://www.espressif.com/en/products/socs/esp32, last accessed 2023/06/10
7. IEEE 802.11 WIRELESS LOCAL AREA NETWORKS https://www.ieee802.org/11/ last accessed 2023/07/20
8. Internet Engineering Task Force (IETF), https://www.ietf.org/, last accessed 2023/06/21
9. Dashboard, https://www.cyberclick.es/numerical-blog/que-es-un-dashboard, last accessed 2023/06/21

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Generación de documentación automática para sistemas embebidos utilizando uModelFactory

Nicolas Tobías Almaraz[1]

[1] Universidad Tecnológica Nacional - Facultad Regional Buenos Aires, CABA 1179, ARG
nalmaraz@frba.utn.edu.ar

Abstract. A partir del trabajo realizado desde el departamento de ingeniería electrónica de la UTN FRBA este equipo de trabajo desarrolló un software llamado uModelFactory. El mismo permite el diseño, depuración y simulación de máquinas de estados finitas tipo Moore. Dicha herramienta fue concebida para ser utilizada a la hora de diseñar sistemas embebidos. La aplicación tiene la finalidad de ser utilizada tanto en un ámbito industrial como en un ámbito educativo. De hecho, la cátedra de la asignatura Informática II decide implementarla como herramienta didáctica. En base a ello es que fue necesario la funcionalidad de documentar los proyectos desarrollados a partir de la aplicación. La idea fue que el usuario tenga la posibilidad de generar un documento editable con toda la información necesaria acerca del proyecto a partir de unos pocos clics. En otras palabras, que el usuario genere un archivo DOCX que cumpla con cierto template y contenga toda la información previamente configurada. La información a escribir implica todo lo referido al desarrollo del sistema embebido, por ejemplo autor, descripción del proyecto, máquinas de estados con sus respectivos diagramas y tablas, listas de variables, eventos y acciones, implementaciones en C, etc.

Palabras clave: máquinas de estado, documentación, software

1. Antecedentes de la propuesta

El software “uModelFactory” [1] se presenta como una herramienta para asistir en la elaboración de diagramas y Máquinas de Estado Finitas (MEF) tipo Moore gobernados por eventos. A partir de trabajos anteriores, se ha analizado el uso de la misma en el ámbito educativo [2], particularmente en la cátedra de Informática II de la Universidad Tecnológica Nacional - Facultad Regional Buenos Aires, como una tecnología que facilite la enseñanza y el aprendizaje del proceso de diseño y diagramado de MEF con complejidad creciente.

En versiones anteriores la funcionalidad de documentar automáticamente estaba resuelta pero no generaba archivos editables. Pues, en un principio generaba una página web HTML en donde se presentaba la información. Dado que el objetivo era poder generar un template para posibles informes referidos al proyecto esta opción no terminaba de cumplir su objetivo, dado que los archivos HTML no son editables.

Es por ello que nace el desarrollo de una mejora en las funcionalidades de documentación. La propuesta fue generar un archivo DOCX o en su defecto ODT de tal modo que el usuario pueda, o bien, utilizarlo directamente tal como lo generó uModelFactory, o bien modificarlo en función de las necesidades del proyecto.

2. Diseño de la Propuesta

El objetivo del informe es documentar el proyecto, es por ello que incluye autor, fecha, descripción del proyecto, introducción teórica, máquinas de estados (cada una de ellas con su respectivo diagrama y tabla) y finalmente las implementaciones en C.

El usuario tiene la posibilidad de configurar la inclusión o no de algunos de los campos mencionados. Además, se tomó la decisión de conservar las funcionalidades de exportar como HTML para darle al usuario otra opción.

3. Resultados

Luego de desarrollar todo lo mencionado en la sección III, el resultado es un nuevo botón en la barra de herramientas (Fig. 1) que brinda la posibilidad de generar documentación.

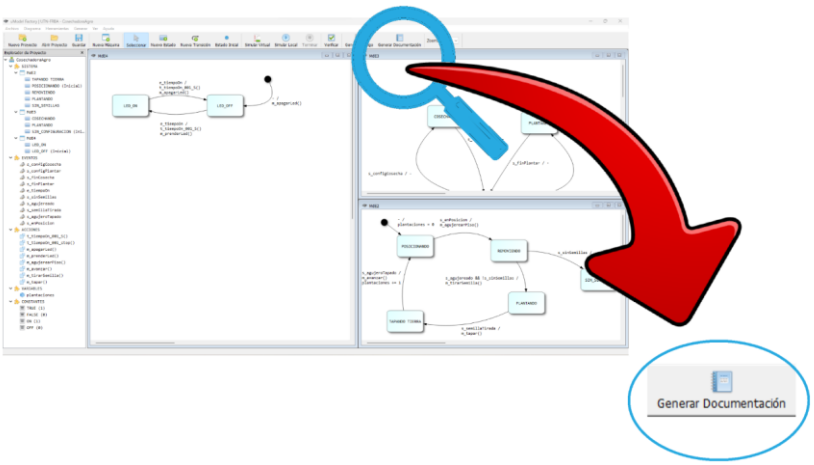


Fig 1. Acceso a funcionalidades desde la barra de tareas.

Este abrirá un menú de configuración en donde el usuario puede configurar qué secciones va a incluir en el informe y si desea exportarlo como DOCX o como HTML (Fig. 2).

Cabe destacar que algunos de los campos se configuran automáticamente, como por ejemplo el nombre del proyecto o el autor (configurado en la creación del proyecto).

Luego, una vez que se configura todo se presiona el botón “Ok” y si todo está bien se abrirá una ventana de confirmación (Fig. 3). En ella indicará la ruta en donde se exportó el archivo y nos facilitará un botón para acceder a dicho directorio.

Finalmente, se obtiene un informe como se detalla a continuación. Particularmente en el ejemplo que se va a desarrollar se configuró sin la introducción teórica y consta de un proyecto de una sola máquina de estados.

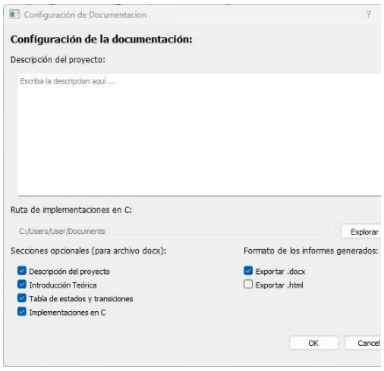


Fig 2. Interfaz de usuario encargada de configurar la documentación automática.

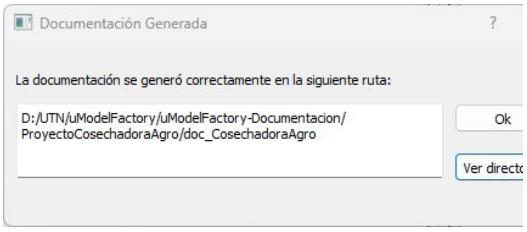


Fig 3. Ventana emergente de confirmación.

En primer lugar, en la Fig. 4 muestra la primera página del informe, en esta podemos ver el encabezado y pie de página con los datos de la universidad y el rótulo de uModelFactory, además de la fecha y datos del autor. También incluye una descripción del proyecto (configurada por el usuario en la Fig. 2) y las listas de acciones, eventos y variables que usa el sistema.

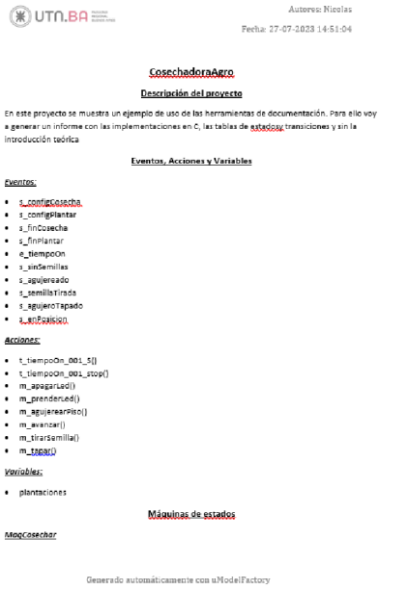


Fig 4. Primera página del informe generado.

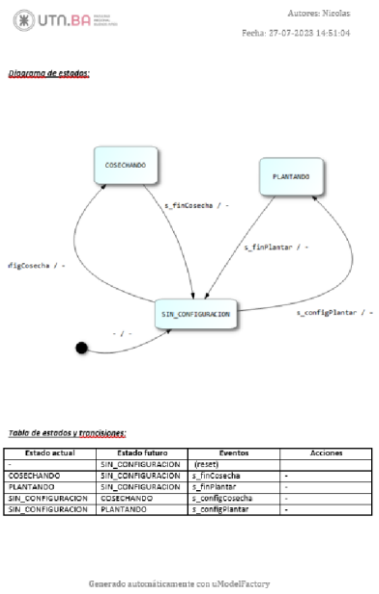


Fig 5. Segunda página del informe.

Por otro lado, la Fig. 5 muestra la segunda página del informe, en esta podemos ver la información referida a la máquina de estados que tiene el proyecto. Aquí se incluye el nombre de la máquina, diagrama y tabla de estados y transiciones. Es importante

aclarar que si el proyecto tuviese otras máquinas van a incluirse una a continuación de la otra hasta terminar de adjuntar la información de todas.

Finalmente, se incorpora una sección del informe que incluye las implementaciones en C configuradas por el usuario. Esta funcionalidad se encargará de cargar el contenido de todos los archivos que se encuentren en el directorio que configuró el usuario en la Fig 2. Cabe destacar que la aplicación tiene implementada la funcionalidad de generación de código automática. Sin embargo, al ser una ruta genérica admite códigos que no fueron exportados directamente desde uModelFactory.

4. Implementación

Todo el proyecto está desarrollado con el framework Qt. Sin embargo, no hay clases desarrolladas para la escritura de archivos DOCX directamente de Qt. Es por ello que la implementación de la escritura se realizó desde Python utilizando el módulo python -docx [4].

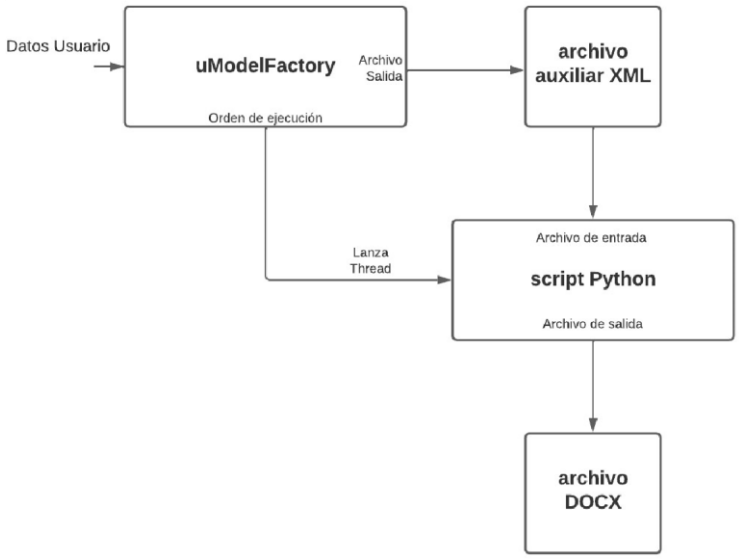


Fig 7. Esquema de funcionamiento.

La estrategia fue escribir un documento auxiliar XML desde la aplicación con toda la información necesaria y luego mediante un nuevo proceso llamar al script de Python para que lea dicho archivo auxiliar y escriba todo lo solicitado (Fig. 7).

Por otro lado, la implementación dentro del bloque uModelFactory se divide en tres niveles jerárquicos principales.

- En el de más alto nivel se arma el template del informe a realizar y recibe como argumentos el contenido del informe
- En el nivel intermedio se hacen métodos para poder escribir más fácilmente el

template, como por ejemplo definir el formato de los encabezados, títulos, subtítulos, etc.
- En el más bajo nivel se encuentran las funcionalidades más básicas como escribir texto con formato, imágenes, tablas, listas, encabezado y pie de página, etc.

5. Conclusiones

Con el desarrollo de este módulo del proyecto se cubren las necesidades planteadas. Es decir, contar con documentación generada a partir del modelo en forma automática, incorporando elementos y recursos de base como también opcionales, en función de la necesidad del usuario o del proyecto. Dicha documentación se encuentra en un formato editable y sirve de base para poder complementar la documentación con otros módulos que se desarrollen o de terceros. En este caso, el módulo de documentación fue también pensado como un aliado para los estudiantes de Informática II, que deben realizar un trabajo práctico integrador, no solo diseñando una solución para un sistema embebido sino también en el soporte para la continuidad de dicho proyecto. En un futuro se tendrá una devolución por parte de los usuarios, cohorte 2023, correspondiente a los estudiantes de la asignatura mencionada, perteneciente al segundo nivel de la carrera de Ingeniería Electrónica de la Universidad Tecnológica Nacional Facultad Regional Buenos Aires.

6. Trabajos a futuro

Como desarrollo a futuro está planificado una serie de mejoras en las funcionalidades. En particular poder seleccionar archivos fuente y elegir qué funciones principales vale la pena poder incluir y cuáles no serán necesarias. En ese sentido, el próximo desafío es poder interpretar el código de los archivos fuente seleccionados para poder permitirle al usuario incluir solamente algunas funciones o solamente algunos de los archivos que se encuentren en la ruta indicada.

Referencias

1. L. Sugezky, M. Prieto, N. González, M. Giura, Y. Kuo, M. Trujillo, J.M. Cruz. "Desarrollo e implementación de herramientas de simulación de modelos para sistemas embebidos". Congreso Argentino de Sistemas Embebidos, 2016.
2. N. González, L. Sugezky, M. Prieto, M. Giura, Y. Kuo, M. Trujillo, J.M. Cruz, “Evaluación del software uModelFactory como herramienta didáctica”. IEEE Argencon, 2016.
3. N. Gonzalez, J. Cruz, L. Sugezky, M. Giura, M. Trujillo, M. Prieto.
4. “Analysis of a UML-based embedded system modeling software application”. Congreso Argentino de Sistemas Embebidos, 2014.
5. python-docx, "python-docx documentation", Versión 0.8.11, python-docx, 2023. Disponible en: https://python-docx.readthedocs.io/en/latest/# . [Consultado el: 20/07/2023].

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Plataforma web para el área de deportes electrónicos

Lorenzo Handula[1]
Luciano Iglesias[1], Cecilia Sanz[1,2]

[1] Instituto de Investigación en Informática LIDI. Facultad de Informática – Universidad Nacional de La Plata

[2] Comisión de Investigaciones Científicas de la Provincia de Buenos Aires

{lhandula,li}@info.unlp.edu.ar, csanz@lidi.info.unlp.edu.ar

Abstract. En este trabajo se presenta el avance de una tesina de grado correspondiente a la Licenciatura en Sistemas de la Facultad de Informática de la Universidad Nacional de La Plata (UNLP). En el marco de este trabajo se aborda el desarrollo de una plataforma vinculada al área de eSport, que se utiliza en el proyecto eSports de la UNLP, y que forma parte de las acciones del Instituto de Investigación en Informática LIDI. Se detallan aspectos claves de la propuesta, algunos de los resultados obtenidos en una sesión de evaluación llevada a cabo con potenciales usuarios, y posibles mejoras a futuro.

Keywords: Deportes electrónicos, Esports, Métricas, Estadísticas,

1. Motivación

Impulsados en gran medida por los avances tecnológicos, la prevalencia del juego online, el acceso a la tecnología y la aparición de grandes competencias, la popularidad de los deportes electrónicos ha crecido enormemente en los últimos años, convirtiéndose en una de las formas más divulgadas de entretenimiento en la actualidad [1].

La audiencia de los principales torneos de deportes electrónicos a menudo supera la de los eventos deportivos tradicionales. Por ejemplo, en 2017, el Campeonato Mundial de League of Legends, el deporte electrónico más prevalente, recibió 60 millones de espectadores, mientras que el evento de básquetbol más importante del mundo, las finales de la NBA, atrajeron solo un tercio de esa cifra: 20,38 millones de espectadores [2]. En el aspecto económico, la industria de los deportes electrónicos generó USD 1200 millones en 2019 [3], superando de hecho los USD 1000 millones que se habían estimado para el año 2020 [4].

A pesar de este crecimiento continuo y exponencial, los deportes electrónicos, o también conocidos como eSports, carecen de una definición generalmente aceptada. Usualmente, se los considera como el equivalente a jugar videojuegos de manera competitiva dentro de un ámbito profesional [5]. En una definición más completa, se los puede entender como un área de los deportes en donde las personas desarrollan y entrenan habilidades mentales y/o motrices en el uso de tecnologías de la información y comunicación [6]. Sin embargo, algunos consideran el término como un oxímoron, por eso se dice que la inclusión de esta actividad dentro del grupo de los deportes tradicionales es aún un tema de debate [7].

Desempeñando su papel de brindar apoyo en las nuevas metas e intereses que van surgiendo entre sus estudiantes, la universidad conformó el proyecto eSports UNLP [8].

Este espacio, dedicado a encabezar todas las actividades relacionadas con la práctica de deportes electrónicos en la institución, es con el cual se trabaja en conjunto para que el desarrollo propuesto satisfaga las necesidades que enfrenta en la actualidad.

Mientras la organización de competencias es una pieza clave a la hora de acrecentar y fortalecer a la comunidad, el análisis de datos brinda un gran cimiento en la formación de un equipo institucional competitivo. Por lo tanto, ante la falta de herramientas tecnológicas que le den soporte de manera específica a estas necesidades, el brindar una solución que combine ambas funcionalidades y, a su vez, ofrezca un portal en donde sus interesados puedan acceder a las últimas novedades de la escena, fue lo que motivó la realización del trabajo.

2. Aporte del trabajo

La tesina aporta a nivel práctico una plataforma web con un panel para la difusión de novedades con respecto al proyecto, y dos funcionalidades principales: la generación de reportes estadísticos avanzados para los equipos institucionales y la gestión de competencias de deportes electrónicos por la universidad. En un principio, para una de las disciplinas más populares actualmente: League of Legends. Está disponible en: https://bitshortly.com/qe49e. Presenta la posibilidad de contar con roles de usuarios. Estos son: Capitán de equipo, Administrador, Autor y Visitante. Cada uno de estos roles tiene acceso a funcionalidades diferentes. Por ejemplo, un capitán de equipo puede cargar los datos de una partida como el tiempo completo, resultado y los jugadores titulares que disputaron el encuentro, y datos estadísticos; un administrador puede generar fixtures, un autor puede crear una novedad, y un visitante puede leer toda la información disponible como novedades (ver Fig.1), la posición actual de un determinado equipo o estadísticas.

Para atacar rápida y eficazmente estas necesidades del proyecto, se optó por seleccionar como base para el desarrollo el CMS Wordpress. Esta herramienta permite el pronto despliegue de un sitio con la operatividad básica necesaria resuelta y con la posibilidad de incorporar recursos con gran diseño sin la necesidad de un equipo de desarrollo. A nivel estructural, se piensa como una instancia general instalada en el dominio raíz, que únicamente posee el panel de novedades y redirecciona al visitante a las distintas instancias para cada disciplina. La Fig. 2 muestra un esquema de la solución.

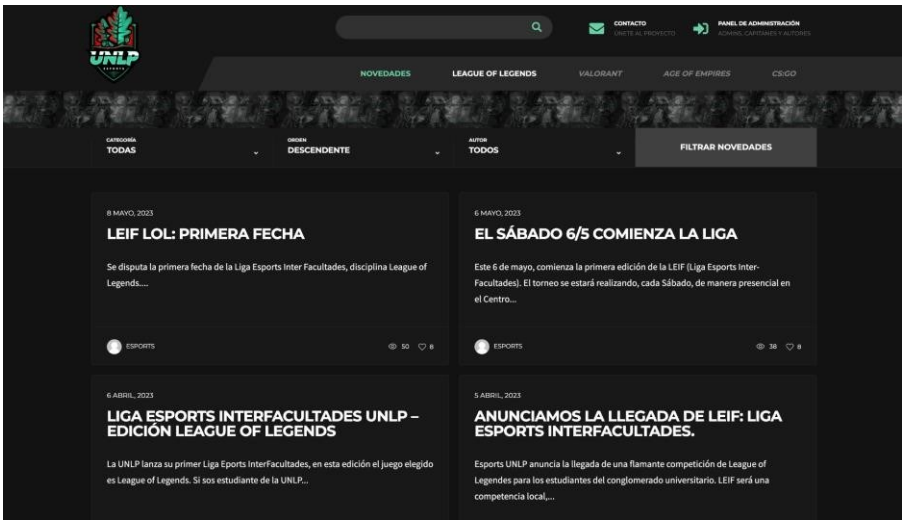


Fig. 1. Panel de novedades. Instancia raíz.

En cuanto a la generación de reportes, tras un análisis de los distintos enfoques posibles estudiados, se llega a una solución gratuita y completamente en la nube gracias a través de los servicios de Google. La recolección se hace a través de un script que se ejecuta diariamente en el servicio llamado App Scripts. Este script desarrollado se comunica mediante la API privada que la desarrolladora del videojuego (Riot) ofrece, procesa los datos de las partidas disputadas por los integrantes del equipo institucional el día anterior y los almacena en un archivo formato .csv en otro servicio gratuito de Google llamado Drive. Finalmente, este archivo es consumido por la herramienta de reportes analíticos en línea de Google llamado Looker Studio como fuente de datos, que gracias a la periodicidad del script, se actualiza cada día de forma automática. Gracias a la capacidad de embeberse en formato *iframe*, este reporte es integrado en la instancia Wordpress del subdominio de League of Legends.

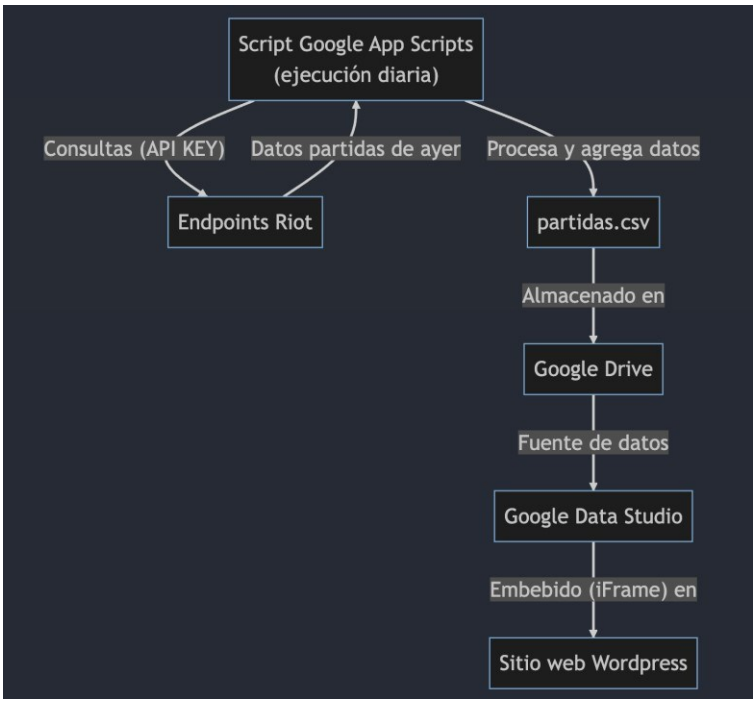


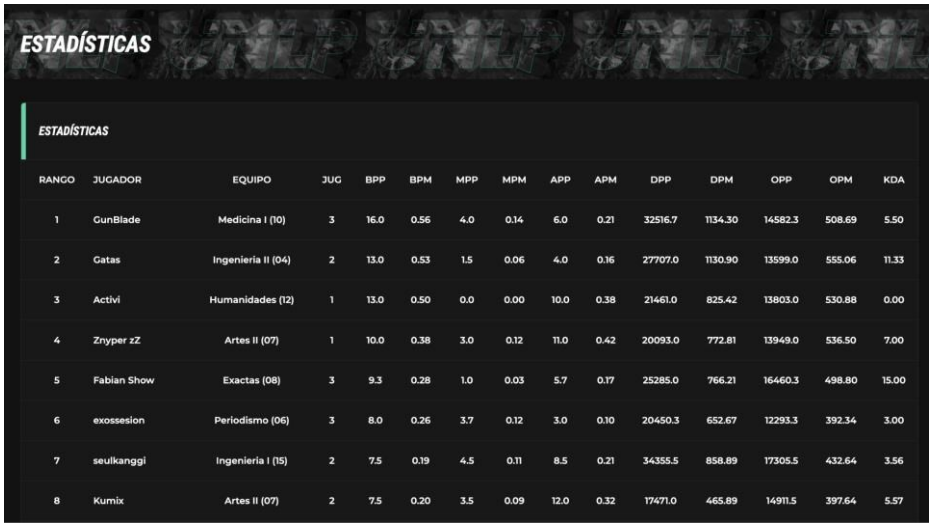
Fig. 2. Esquema solución recolección de datos y reporte de métricas

El reporte generado puede ser modificado en cualquier momento por el capitán del equipo institucional y permite analizar decenas de métricas individuales (ver Fig 3 a Fig. 4) de los jugadores como: porcentaje de oro recolectado con respecto a sus compañeros, promedio de bajas, asistencias, muertes, puntos de visión y más. Además, todos los datos pueden ser filtrados por jugador, parche, campeón utilizado, campeón rival, victoria/derrota, posición en el campo de juego y fecha.



Fig. 3. Reporte de métricas del equipo institucional

Al mismo tiempo, se ha trabajado en la integración de una herramienta para la gestión de competencias. Gracias a esta herramienta, la competencia interfacultades LEIF puede desarrollarse por completo, incluyendo: generación de fixtures, tablas de posiciones, detalle de partidas, estadísticas individuales y más.



ESTADÍSTICAS

ESTADÍSTICAS

RANGO	JUGADOR	EQUIPO	JUG	BPP	BPM	MPP	MPM	APP	APM	DPP	DPM	OPP	OPM	KDA
1	GunBlade	Medicina I (10)	3	16.0	0.56	4.0	0.14	6.0	0.21	32516.7	1134.30	14582.3	508.69	5.50
2	Gatas	Ingeniería II (04)	2	13.0	0.53	1.5	0.06	4.0	0.16	27707.0	1130.90	13599.0	555.06	11.33
3	Activi	Humanidades (12)	1	13.0	0.50	0.0	0.00	10.0	0.38	21461.0	825.42	13803.0	530.88	0.00
4	Znyper zZ	Artes II (07)	1	10.0	0.38	3.0	0.12	11.0	0.42	20093.0	772.81	13949.0	536.50	7.00
5	Fabian Show	Exactas (08)	3	9.3	0.28	1.0	0.03	5.7	0.17	25285.0	766.21	16460.3	498.80	15.00
6	exossesion	Periodismo (06)	3	8.0	0.26	3.7	0.12	3.0	0.10	20450.3	652.67	12293.3	392.34	3.00
7	seulkanggi	Ingeniería I (15)	2	7.5	0.19	4.5	0.11	8.5	0.21	34355.5	858.89	17305.5	432.64	3.56
8	Kumix	Artes II (07)	2	7.5	0.20	3.5	0.09	12.0	0.32	17471.0	465.89	14911.5	397.64	5.57

Fig. 4. Líderes estadísticos de la LEIF.

Así la solución desarrollada se compone de distintas tecnologías que fueron siendo estudiadas como partes de la tesina de grado mencionada y se han integrado dando

forma a la plataforma de eSports propuesta. En conjunto funcionan de manera transparente para el usuario visitante (ver Fig.2).

3. Evaluación, Conclusiones y Líneas de investigación futura

Es importante destacar que se ha llevado a cabo una sesión de pruebas con 8 usuarios, entre los que se contó con personas vinculadas a los eSports (relacionadas con el equipo de la UNLP) y otras que no tenían conocimiento. Se presentaron escenarios de prueba con tareas que cada usuario debió desarrollar utilizando un usuario de prueba con un rol específico. Se contempló para cada tarea solicitada a cada usuario si fue cumplida o no y los tiempos empleados para llevarlas a cabo. En forma posterior, se realizó un focus group donde los participantes realizaron comentarios sobre las tareas realizadas y su dificultad, y sobre aspectos de usabilidad de la plataforma. Los resultados indican que todas las tareas solicitadas fueron realizadas por los participantes. Los usuarios sin conocimiento en el tema de eSports, completaron las tareas en un tiempo mayor a los que tenían conocimiento e hicieron preguntas sobre los significados de los indicadores estadísticos. Los participantes que tuvieron el rol de Administradores de Liga fueron los que más tardaron en completar el escenario de prueba, ya que las tareas requerían más pasos, el resto lograron terminarlo en tiempos similares y sin mayores dificultades. En el focus group la plataforma fue valorada como útil y fácil de usar, pero con necesidad de conocer algunos indicadores estadísticos por parte de los usuarios sin conocimiento.

El próximo desafío técnico que enfrenta la plataforma es la automatización del proceso de carga de datos correspondientes a las partidas de la LEIF. Hoy en día es un proceso manual, pero en un futuro podría realizarse automáticamente mediante la misma API utilizada en la generación de reportes, donde en lugar de obtener los datos de todas las partidas para un grupo de jugadores, se obtendría para una partida específica. Al mismo tiempo se realizarán pruebas de usabilidad.

Además, se espera que toda la implementación pueda reproducirse para el resto de disciplinas que integran al proyecto Esports UNLP: CS:GO, FIFA, Age of Empires y Valorant.

Referencias

1. Hamari, J., & Sjöblom, M. (2017). What is esports and why do people watch it?.
2. Steinkuehler, C. (2019). Esports research: Critical, empirical, and historical studies of competitive videogame play. Games and Culture.
3. Gawrysiak, J., Burton, R., Jenny, S., & Williams, D. (2020). Using esports efficiently to enhance and extend brand perceptions - a literature review. Physical culture and sport studies in research.
4. NewZoo. (2020). Global esports market report 2020, 14th July 2020. https://bit.ly/newzoo2020.
5. Welch, T. (2002). "The History of the CPL", Cyberathlete Professional League.
6. Wagner, M. G. (2006). On the Scientific Relevance of eSports.

7. Witkowski, E. (2012). On the digital playing field: How we “do sport” with networked computer games. Games and Culture.
8. Proyecto eSports – Universidad Nacional de La Plata. Disponible en: https://esports.info.unlp.edu.ar/

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Desarrollo de un modelo con XGBoost intérprete de la Lengua de Señas Argentina

Facundo Recabarren, Fabrizio Amaya, Raúl Klenzi, María Isabel Masanet

Universidad Nacional de San Juan, San Juan 5400, Argentina
{facunicolasrecabarren, fabrizio.amaya88, rauloscarklenzi, mimasanet}@gmail.com

Resumen. Se propone el desarrollo de un modelo con XGBoost intérprete de la Lengua de señas Argentina (LSA), encontrando la correspondencia entre una secuencia de imágenes (video) y su apropiado texto en español. Para esto se plantea un modelo en árboles de decisión optimizado aplicado a un conjunto de datos preexistentes sobre el que se realizan tareas de preprocesamiento, muestreo y almacenamiento.

Palabras clave: Lengua de Señas Argentina, LSA, Machine Learning, Data Science, MediaPipe, XGBoost,

1. Introducción

Las lenguas de señas son lenguas que utilizan la modalidad visual-gestual para transmitir significado a través de articulaciones manuales en combinación con elementos no manuales como el rostro y el cuerpo. Sirven como el principal medio de comunicación para numerosas personas sordas y con problemas de audición. Al igual que los lenguajes orales, los lenguajes de señas son lenguajes naturales regidos por un conjunto de reglas lingüísticas y a diferencia de aquellos, la Lengua de Señas de un idioma en particular, no es una forma visual de ese idioma, sino que es un lenguaje propio. El procesamiento del lenguaje de señas es un campo emergente de la inteligencia artificial, centrándose su investigación en los aspectos visuales de estos lenguajes, cubriendo un subcampo tanto del procesamiento de lenguaje natural (NLP) como de la visión por computadora (CV). https://research.sign.mt/

La Comunidad Sorda es un grupo minoritario que lentamente ha logrado integrarse en las comunidades de todo el mundo. Si bien en algunos países su integración está muy avanzada, en nuestro país aún queda mucho por hacer para lograrlo.

Las personas sordas hacen un gran uso de la tecnología, priorizando el contexto "visual" debido a que la lengua de señas se entiende por el sentido de la vista. Es así que utilizan videollamadas, mensajes de texto, teléfonos móviles, notebooks y todo aquello que les permite comunicarse y darse a entender.

San Juan carece de los medios y tecnologías adecuadas para ellos, y no se cuenta con intérpretes de LSA en casi ningún ámbito, ya sea hospitales, bancos, edificios públicos, incluso tampoco los hay en escuelas, universidades y demás.

Este trabajo presenta el desarrollo de un modelo que interpreta señas de la Lengua de Señas Argentina (LSA) y forma parte de una tesis de Licenciatura en Ciencias de la Computación en ejecución.

2. Obtención de Datos

Para el entrenamiento de los distintos modelos a analizar es necesario contar con un Set de Datos con el cual llevar a cabo ese entrenamiento. Este Set de Datos puede construirse a medida para el problema en particular o también, en caso de ser posible, usar un Set de Datos ya construido y que pueda ser aplicado al contexto del problema.

En esta propuesta se hace uso del Set de Datos "LSA64" para llevar a cabo el entrenamiento, este Set de Datos fue obtenido como parte de los resultados de la Tesis Doctoral de Franco Ronchetti en [1] y se permite la utilización del mismo en [2].

2.1 Características del Set de Datos

El Set de Datos cuenta con un conjunto de 64 señas interpretadas, cada una de ellas, por 10 personas las cuales, a su vez, grabaron 5 videos para cada seña, obteniendo un total de 3200 videos (50 videos por cada seña). La particularidad de este set de videos, es que los intérpretes utilizan un guante de color diferente por mano (rosado en la mano derecha y verde en la izquierda), siendo en este caso la mano derecha la dominante.

"La base de datos fue construida en dos sets de grabación distintos. En el primero fueron grabadas 23 señas con una sola mano y se utilizó luz natural en un entorno abierto. En el segundo set, se agregaron 41 señas más (22 con dos manos, y 19 con una mano) y se utilizó luz artificial en un entorno semi-cerrado…. La base de datos es públicamente accesible bajo licencia Creative Commons Attribution-NonCommercialNoDerivatives 4.0 International sólo para uso académico. Se encuentra almacenada en https://midusi.github.io/lsa64/index.html." [2]

Otra característica de los videos es que estos fueron capturados en calidad FullHD o 1920x1080 y a 60 fotogramas por segundo (FPS).

De la lista original de 64 señas se han tomado un subconjunto de 21 señas que se realizan con una mano con el objetivo de poder formar palabras con sentido.

3. Procesamiento de videos

Se utiliza un framework de Machine Learning llamado "MediaPipe" [3] el cual es capaz de procesar videos e imágenes entregando como resultado la detección de personas en la imagen recibida como entrada.

La salida que entrega el modelo es una secuencia de coordenadas conocida como skeletal data. Estos datos se encuentran normalizados dentro del intervalo [0.0, 1.0] con respecto al alto y ancho de la imagen a procesar.

Dentro de este framework se encuentran distintos modelos pre-entrenados y listos para usar, utilizándose en este caso el llamado "Holistic Tracking" dado que permite obtener una estimación de puntos corporales correspondientes a ambas manos, la cara y la posición corporal con un único modelo.

3.1 Preprocesamiento

Los videos originales atraviesan una primera etapa de procesamiento en la cual se obtiene de cada video un equivalente de los frames o imágenes que lo componen, pero representado por una lista de puntos corporales asociados a cada frame detectado por el modelo holístico. Así, cada video se representa por una lista de puntos corporales o coordenadas para la persona detectada en cada frame, de suma relevancia dado que el algoritmo XGBoost se destaca por sobre otros, cuando procesa datos tabulares [4] [5].

3.2 Muestreo o Sampling

Una vez obtenidos los datos numéricos a través del preprocesamiento de los videos y a los efectos de reducir la tasa de datos a procesar, se procede a capturar ciertos frames de forma aleatoria o siguiendo alguna técnica empírica específica.

Seguidamente, se detallan 3 alternativas para acotar los videos a 10 y 30 frames (X):

- Muestreo Random: Se toma para cada video una cantidad de X frames aleatoriamente, siendo estos frames tomados del conjunto de frames/imágenes que componen cada video.
- Muestreo Random con Porcentajes: En este caso se separa cada video en 3 secciones: inicial, intermedia y final. La sección inicial está compuesta por frames aleatorios provenientes del primer 20% del video, luego el siguiente 60% de frames conforma la sección intermedia y el 20% restante representa la sección final. De cada sección se tomará la cantidad necesaria de frames de manera de poder completar el primer 20% de X, luego 60% y 20% restante. Por ejemplo: para un muestreo de 10 frames, se tomarán 2 frames de la sección inicial, 6 frames de la sección intermedia y por último 2 frames de la sección final.
- Muestreo Random con Poda: El tercer muestreo realiza una poda de frames del 15% de frames los iniciales y finales, es decir, una poda del 30%, exceptuando aquellos videos donde al quitar ese 30% la cantidad de frames restantes no es suficiente para completar la cantidad de X frames necesarios.

Las 3 técnicas se utilizaron para armar sets de datos acotados a 10 y 30 frames, además, se armaron sets de datos con y sin datos faciales con el objetivo de analizar su performance. Como salida de este paso, se obtuvieron 12 sets de datos, resultantes de la combinación de los distintos muestreos realizados.

4. Entrenamiento

El siguiente paso consiste en la construcción, entrenamiento y optimización del modelo sobre los distintos sets de datos obtenidos en el paso previo. El modelo elegido está construido con XGBoost (Xtreme Gradient Boosting) la cual "es una librería de optimización del gradiente distribuida y optimizada para ser altamente eficiente, flexible y portátil" [4].

4.1 XGBoost

XGBoost proviene del inglés “Extreme Gradient Boosting” o “Refuerzo extremo del gradiente” es una herramienta desarrollada para entrenar mediante el aprendizaje supervisado árboles de decisión. Cada árbol que se genere aprenderá de sus predecesores e intentará disminuir el error incurrido en ellos.

El funcionamiento puede describirse como:

“Se obtiene un árbol inicial F0 para predecir la variable objetivo ‘y’, el resultado se asocia con un residual (y - F0).

Se obtiene un nuevo árbol h1 que ajusta el error del paso previo.

Los resultados de F_0 y h_1 se combinan para obtener el árbol F_1, donde el error cuadrático medio de F_1 será menor que el de F_0:

$$F_1(x) = F_0(x) + h_1(x)$$

Este proceso se sigue iterativamente hasta minimizar el error de la siguiente forma:

$$F_m(x) = F_{m-1}(x) + h_m(x)$$” [5]

Los datos obtenidos en la fase de preprocesamiento se utilizan como entrada directa para este modelo, trabajando con datos tabulares .

4.2 Evaluación

Los sets de datos construidos anteriormente son divididos en 3 subconjuntos: entrenamiento, validación y testeo. Luego de realizar un primer entrenamiento se procede a una siguiente etapa de “Optimización de Hiperparámetros” con el objetivo de encontrar una combinación de valores que eleve el rendimiento de cada modelo.

Los hiperparámetros evaluados son: learning_rate, n_estimators, max_depth, min_child_weight, subsample, colsample_bytree, reg_alpha y reg_lambda.

Durante el proceso de entrenamiento se hace uso de técnicas como cross_validation. Para evaluar los resultados que presenta el modelo se seleccionaron las siguientes métricas: Precision, Recall, F1-score, F05-score, F2-score y Accuracy.

En Tabla 1 se presentan los resultados obtenidos sobre 2 sets de datos, con distinta técnica de muestreo, antes y luego de haber realizado el proceso de optimización:

Tabla 1. Resultados de entrenamiento para subconjunto de testeo en modelo base y optimizado.

Métrica/Muestreos	**criterio Random 30 frames sin cara**		**criterio Proporción 30 frames con cara**	
	Base	Optimizado	Base	Optimizado
Precision	0.93	0.96	0.95	0.95
Recall	0.92	0.95	0.93	0.95
F1	0.92	0.95	0.93	0.95
F05	0.93	0.95	0.94	0.95
F02	0.92	0.95	0.93	0.95
Accuracy	0.92	0.95	0.93	0.95

5. Conclusión

La etapa de entrenamiento al ejecutar el algoritmo XGBoost con cross-validations demora entre 5 y 8 minutos. El equipo donde se trabajó posee los siguientes componentes: Intel I5 8va generación, GPU GTX 1050 2GB, aunque también actualmente se están realizando pruebas sobre Google Colab.

La Tabla1. muestra como mejoran las métricas tras la optimización de hiperparámetros. Puede observarse cómo el proceso aumenta en un pequeño porcentaje (0.01-0,03) el rendimiento, siendo este rendimiento evaluado en base a la seña que el modelo interpreta para cada video de entrada no procesado previamente (subconjunto de testeo). La elección de los sets de datos para Tabla1. está basada en aquellos que presentaron un mayor porcentaje de mejora en el proceso de optimización de hiperparámetros. De ser necesaria la elección de un modelo final, en base a lo observado en Tabla1., el modelo construido sobre el set de datos "criterio Random 30 frames sin cara" sería el elegido, debido a que, con menor cantidad de información (no se cuenta con datos faciales), se logran resultados similares y/o iguales.

Actualmente se están analizando modelos construidos con capas LSTM/GRU pertenecientes a Redes Neuronales Recurrentes de los cuales próximamente se contarán con los resultados pertinentes y se realizarán comparaciones que resulten necesarias.

Así mismo, con la asistencia de intérpretes de LSA se pretende generar desde el set de videos originales una mayor cantidad de secuencias de imágenes con frases coherentes y con ello lograr reconocer los inicios y finales de cada palabra en la secuencia.

Referencias y Bibliografía

1. Ronchetti, Franco: Reconocimiento de gestos dinámicos y su aplicación al lenguaje de señas. Tesis Doctoral, Facultad de Informática (2017).
2. Ronchetti, Franco and Quiroga, Facundo and Estrebou, Cesar and Lanzarini, Laura and Rosete, AlejandroLSA64: A Dataset for Argentinian Sign Language. Recuperado el 01 de Agosto del 2023 de http://facundoq.github.io/datasets/lsa64/
3. MediaPipe Framework Recuperado el 04 de Agosto del 2023 de https://developers.google.com/mediapipe/framework
4. XGBoost. Recuperado el 04 de Agosto del 2023 de https://xgboost.readthedocs.io/en/stable/
5. Espinoza, J.: Aplicación de algoritmos Random Forest y XGBoost en una base de solicitudes de tarjetas de crédito. Ingeniería Investigación y Tecnología volumen XXI, 3 (2010).
6. Sign Language Procesing. Recuperado el 01 de Agosto del 2023 de https://research.sign.mt/ Sai B. Padigala, Gogineni H. Madhav, Saranu K. Kumar, Dr. Narayanamoorthy M: Video based sign language recognition using CNN-LSTM. IRJET, pp 1658-1663 (2022)
7. Ko, S.-K., Kim, C. J., Jung, H., & Cho, C.: Neural Sign Language Translation Based on Human Keypoint Estimation. MDPI, pp 2683 (2019) http://dx.doi.org/10.3390/app9132683
8. DONGXU LI, Cristian Rodriguez, Xin Yu, HONGDONG LI; Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp 1459-1469 (2020).
9. Zhou, Z., Chen, K., Li, X. et al. Sign-to-speech translation using machine-learning-assisted stretchable sensor arrays. Nat Electron 3, pp 571–578 (2020). https://doi.org/10.1038/s41928-020-0428-6
10. HCDPS. Leyes Provinciales N° 7412 (2003) y N° 761-S (2014).

CACiC 2023
CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Modelos de Machine Learning para Vehículos Autónomos aplicados a la Detección de Obstáculos en Centros de Logística

Débora Denise Altamirano Medina[1], Augusto Guillermo Tapparelli[1], Matías Kreder[1] y Pablo Ezequiel Inchausti[1]

[1] Universidad Argentina de la Empresa (UADE), Instituto de Tecnología (INTEC). Buenos Aires, Argentina
{daltamirano, atapparelli, mkreder, pinchausti}@uade.edu.ar

Abstract. Los autos autónomos cada día están tomando mayor importancia y su utilización se ha extendido a varias industrias. Un ejemplo del impacto que se puede lograr por medio de robot automatizados puede observarse en Amazon y la operación automatizada en sus centros de almacenamiento y distribución de mercadería. Pero realizar investigaciones asociadas a vehículos autónomos requieres de un ecosistema complejo, con simuladores virtuales, vehículos a escala, entornos controlados y una adecuada configuración de dispositivos que permita la experimentación sobre una versión simplificada de la realidad. En ese contexto, se propone utilizar a la plataforma de AWS DeepRacer con sus entornos físicos y virtuales, como medio de investigación de soluciones para los centros de logística. Se proponen tres objetivos operativos con distinto grado de soporte nativo en la plataforma, estos son en los centros de logística: esquivar obstáculos, desplazar objetos y alertar sobre líquidos. Para el entrenamiento de los modelos de machine learning se propone utilizar aprendizaje por refuerzo.

Keywords: Vehículos Autónomos, Machine Learning, Reinforce Learning.

1. Introducción

Con el avance de la tecnología, la posibilidad de que las máquinas actúen de forma autónoma tiene cada vez más relevancia. En el caso de la industria automotriz, el avance de la inteligencia artificial y la eficiencia en la velocidad de procesamiento de los dispositivos ha logrado que disminuir la necesidad de la interacción humana a tal punto que la posibilidad de autos completamente autónomos ya son una realidad.

Se entiende como autos autónomos a vehículos que tienen la capacidad de percibir el entorno que lo rodea y navegar a través del ambiente de forma segura con poca o sin interacción humana [1], y, de acuerdo al grado de autonomía del vehículo, *SAE International* definió una escala de cinco niveles, en dónde el Nivel 0 carece de automatizaciones con el conductor realizando todas las acciones, y el Nivel 5 completamente automatizado en dónde el conductor solamente establece el destino y el vehículo realiza todas las acciones sin restricciones. [2]

Para lograr el grado de autonomía de Nivel 5, los vehículos incorporan sensores propioceptivos, es decir, del su estado interno, como la fuerza, energía, voltajes, carga, radio de ángulos, etc. Por ejemplo, utilizando los sistemas de navegación (GNSS y GPS), las unidades de medida inercial (IMU) y los acelerómetros, giroscopios y magnetómetros. También disponen de sensores exteroceptivos, es decir, de su estado

externo, como ser cámaras, radares, sensores ultrasónicos y el LiDAR que permite detectar obstáculos por medio de la emisión de un láser [1].

Entre los casos de uso más difundidos que buscan garantizar la seguridad de los vehículos autónomos se encuentran los recorridos de buses autónomos sobre circuitos controlados, y los centros de logística [3]. Se puede destacar el grado de automatismo en los almacenes de la tienda Amazon, con el robot *Proteus* completamente autónomo capaz de desplazar cargas, y operar no solo dentro de una zona de seguridad, sino que es capaz de compartir el área de trabajo con las personas y esquivarlas [4].

En el presente trabajo se investiga sobre el desarrollo de modelos de *machine learning (ML)* para vehículos autónomos, y considera a los centros de logística como el escenario de aplicación.

Para realizar el desarrollo sobre un entorno simulado físico y un entorno virtual se propone utilizar la plataforma AWS DeepRacer de Amazon que ofrece vehículos autónomos a escala 1:18 equipados con los dispositivos y sensores necesarios, entre ellos las cámaras y el LiDAR. [5]

El presente trabajo se corresponde al Proyecto Final de Ingeniería Informática (PFI) de Débora Altamirano y Augusto Tapparelli en curso durante 2023 y se integra dentro del INTEC, el Instituto de Tecnología de UADE, al proyecto de investigación *A21T08 Aplicaciones de Machine Learning y IoT sobre Vehículos Autónomos.*

1.1 Objetivos

Se define como el objetivo principal, desarrollar modelos de *Machine Learning* para permitir a un vehículo autónomo realizar diversas funciones aplicables a centros de logística, para el territorio argentino en el año 2023.

Y para lograr el objetivo principal se propone, como objetivos específicos, desarrollar los siguientes tres modelos de ML:

- Modelo de ML para identificar y esquivar obstáculos.
- Modelo de ML que permita identificar y desplazar objetos.
- Modelo de ML que permita identificar líquidos y notificar.

1.2 Alcance

En términos de alcance, se propone utilizar la plataforma AWS DeepRacer para probar el desempeño de los modelos de *machine learning*, utilizando tanto el entorno virtual para evaluar los modelos, como el entono físico utilizando los vehículos AWS DeepRacer EVO y la pista con el circuito de recorrido y la posibilidad de incorporar obstáculos en el camino [5]. Respecto al método de entrenamiento, se utiliza en los modelos de ML el aprendizaje por refuerzo, con funciones de recompensa con la premisa de maximizar el premio para el agente si se acerca al objetivo.

2. Materiales y Métodos

Para el desarrollo de los modelos de ML y los algoritmos de aprendizaje por refuerzo, se utiliza la plataforma AWS DeepRacer, tanto en el mundo físico, con el vehículo AWS DeepRacer EVO y la pista de entrenamiento, y los simuladores virtuales. El vehículo AWS DeepRacer EVO está equipado con dos cámaras de 4MP para su visión periférica, cuenta con sensores de aceleración y el giroscopio, y el sensor LiDAR para la identificación de objetos. Respecto a su CPU, tiene un procesador *Intel Atom* con 4GB de memoria RAM y 32 GB de almacenamiento para alojar y ejecutar los modelos entrenados que controlan el dispositivo [6]

Para la simulación en el entorno físico, se utiliza el modelo de pista denominado *A to Z Speedway* con dimensiones de 5.48m x 8.22m [7], y que puede observarse desplegada en la Fig. 1 junto a uno de los vehículos utilizados. La impresión de la pista se realizó con la gestión del departamento de informática, sobre una malla de vinilo con colores mate para minimizar el brillo. También se colocan las barreras laterales, recomendado de color azul para que el vehículo identifique el fin de la pista.



Fig. 1: Vehículo AWS DeepRacer Evo y pista en las instalaciones de UADE

Respecto al entorno virtual, se utiliza el simulador provisto por la plataforma AWS DeepRacer, que permite seleccionar distintos tipos de circuitos y además incorporar obstáculos dentro del recorrido para generar distintos escenarios de entrenamiento.

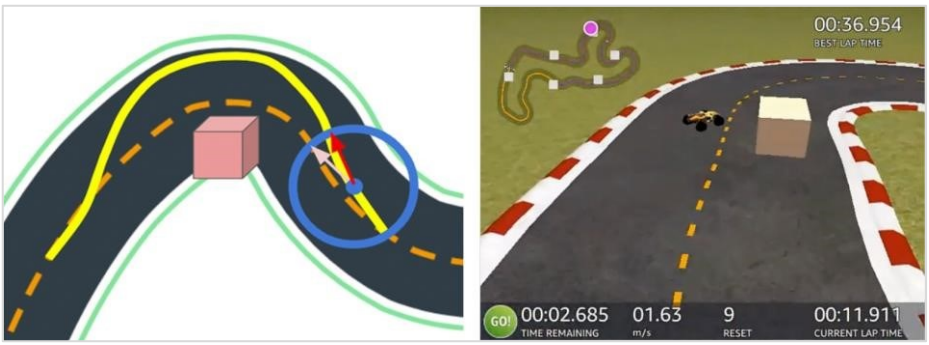


Fig. 2: Emulador virtual en la plataforma AWS DeepRacer con obstáculos [8]

2.1 Entrenamiento de los Modelos de Machine Learning

Para la fase de entrenamiento de los modelos de ML, se utiliza la plataforma *Deep Racer For Cloud (DRFC)* que permite entrenar los modelos de ML de DeepRacer en la nube, localmente en Windows, Linux o Mac. [9]

El entrenamiento se realiza sobre instancias EC2 *spot* de AWS. Las instancias *spot* utilizan capacidad ociosa de AWS y permiten reducir los costos de uso al mínimo. La instancia *spot* utilizada es una *c5.2xlarge* a un costo de usd 0,34 por hora. Se prevé utilizar en el futuro instancias *spot* EC2 con GPU de mayores prestaciones, aunque también mayores costos. En la Fig. 3 se observa una demostración de *Deep Racer For Cloud (DRFC)* realizada a los alumnos de PFI con tres *workers* para el entrenamiento.

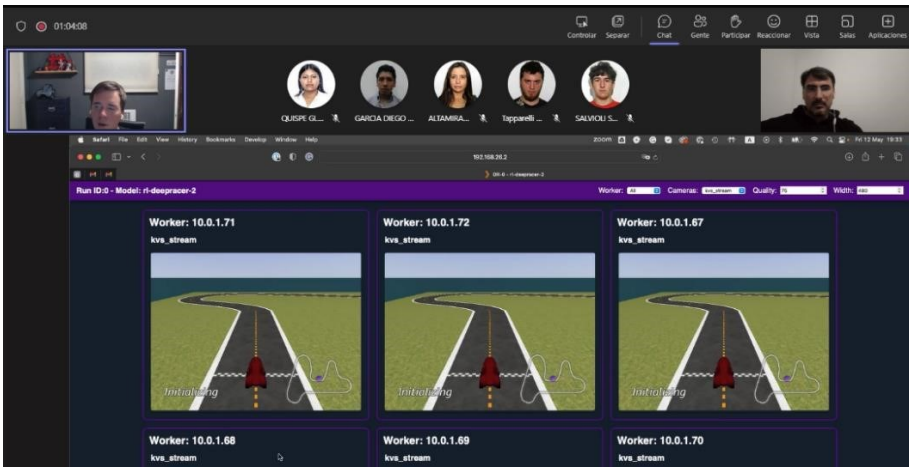


Fig. 3: Demostración del uso de *Deep Racer For Cloud (DRFC)* con alumnos de PFI UADE

Para al desarrollo de los modelos de ML, se utiliza *Python* para programar los algoritmos de aprendizaje por refuerzo, alineados a los tres objetivos específicos propuestos: capacidad de esquivar obstáculos, desplazar objetos y alerta de líquidos.

En el aprendizaje por refuerzo, un agente inteligente realiza acciones en su entorno que son premiadas o no por una función de recompensa, en la medida que el estado del agente se acerca su objetivo.

En el caso de uso de esquivar obstáculos, la plataforma AWS DeepRacer ofrece soporte para el desarrollo y evaluación de las funciones de recompensa. Y en el contexto del presente trabajo se desarrollarán métodos de optimización de modelos. En los casos de uso de desplazar objetos, y detección de líquidos, la plataforma de AWS DeepRacer no ofrece soporte nativo, con lo cual se prevé desarrollar funciones de recompensa que se integren a modelos de reconocimientos de imágenes para identificar tanto los líquidos con el fin de generar una alerta, como los objetos desplazables, con el fin de iniciar una acción de movimiento y traslado a destino.

En el presente estado de avance del PFI, que se encuentra al presente en curso, los detalles de la problemática de desplazamiento de objetos y detección líquidos se encuentran en investigación y desarrollo.

3. Conclusiones y discusión

La investigación aplicada a modelos de ML para vehículos autónomos ofrece la oportunidad de mejorar la eficiencia de los procesos productivos de la industria. Entre los campos de aplicación, los almacenes y centros de logística se presentan como un sector estrechamente relacionado con la automatización de sus operaciones. En particular, la investigación sobre vehículos autónomos requiere de entornos controlados para garantizar la seguridad. También, al simplificar la realidad es posible focalizar en la construcción de procesos escalables de aprendizaje automatizado. En ese contexto de necesidad de entornos físicos o virtuales, la plataforma AWS DeepRacer se presenta como una alternativa por el equipamiento inicial ofrecido, desde los simuladores hasta los vehículos autónomos completamente equipados. Es cierto que la plataforma se orienta a la competición en velocidad, y la falta de soporte nativo hace complejo alcanzar ciertas acciones. Pero la integración de la plataforma AWS DeepRacer con otras tecnologías de ML, como el reconocimiento de imágenes, habilita un nuevo campo para la construcción y el diseño de soluciones.

Agradecimientos. A la Facultad de Ingeniería y Ciencias Exactas (FAIN) y al Instituto de Tecnología (INTEC) de la Universidad Argentina de la Empresa (UADE) por apoyo a la línea de investigación de "*Aplicaciones de Machine Learning y IoT sobre Vehículos Autónomos*" (A23T08).

Referencias

1. H. Ignatious, H. Sayed, y M. Khan, "An overview of sensors in Autonomous Vehicles", Procedia Computer Science, vol. 198, pp. 736-741, (2022) doi:10.1016/j.procs.2021.12.315.
2. SAE International, "Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles". (2021). Accedido: 24/07/2023. https://www.sae.org/standards/content/j3016_202104/.
3. D. Castelló, "Uso de vehículos autónomos y conectados", (2021). Accedido: 24/07/2023. https://dallocas.blogs.upv.es/2021/04/25/uso-de-vehiculos-autonomos-y-conectados/
4. L. Alcalde. (2022) "Amazon Proteus: así funciona su primer robot totalmente autónomo" https://www.businessinsider.es/amazon-proteus-funciona-primer-robot-totalmente-autonomo-1081681. Accedido: 08/08/2023.
5. Amazon Web Services, Inc., "¿Qué es AWS DeepRacer?", (2023). Accedido: 07/08/2023. https://docs.aws.amazon.com/es_es/deepracer/latest/developerguide/what-is-deepracer.html
6. Amazon Web Services, Inc., "AWS DeepRacer, la manera más rápida de iniciarse en aprendizaje automático", https://aws.amazon.com/es/deepracer. Accedido: 07/08/2023.
7. Amazon Web Services, Inc., "AWS DeepRacer track design templates". https://docs.aws.amazon.com/deepracer/latest/developerguide/deepracer-track-examples.html
8. B. Turner, "Bandwidth R&D's AWS DeepRacer 2021 Season Recap", Bandwidth, (2022). https://www.bandwidth.com/blog/bandwidth-rd-aws-deepracer-2021-season-recap/ aws-deepracer-community, "Deepracer-for-Cloud", (2023) https://aws-deepracer-community.github.io/deepracer-for-cloud/. Accedido: 05/08/2023.

Índice de Autores

A

Aballay, L. N. 808
Acosta, N. 205
Alegretti, A. 382
Alfonso, H. 72
Almaraz, N. T. 854
Altamirano Medina, D. D. 871
Álvarez, A. 814, 848
Alvez, C. E. 67
Amandi, A. 205
Amaya, F. 866
Ambrosio, G. 814
Anriquez-Atia, W. M. 711
Antonelli, L. 352, 382
Aragón, V. 227
Asteasuain, F. 279
Aubin, V. I. 217

B

Bandieri, F. 797
Banega, M. 236
Bankoff, K. P. 365
Barrionuevo, M. 160
Battaglia, N. 182, 195
Bazán, P. 11, 730, 764
Bejarano, M. 662
Bellino, F. 11
Beltran, V. 476
Benítez Galván, C. A. 88
Bermudez, C. 72
Bertín, M. I. 342
Bertone, R. 299, 377, 388, 418, 552
Bianco, S. 720
Blanco, G. 342
Bond, R. 531
Brandan, F. 843
Bria, O. 514
Buccella, A. 308

C

Cabrera, S. A. 745
Cagnina, L. 114
Calabro, J. P. 524
Calderón, S. 489
Camele, G. 97
Campetella, M. 308
Campo, M. 205
Candotti, E. 2
Cao, A. 428
Carballo, M. A. 467
Carmona, A. 848
Carnero, M. 48
Carranza, A. 2
Carruthers, J. A. 289
Casas, S. 332
Catania, C. A. 124, 591
Cechich, A. 308
Challiol, C. 398
Chavez, S. B. 451
Cherencio, G. R. 106
Chichizola, F. 489
Cochella, L. 77
Company, A. M. 248
Corbalán, L. 318
Costaguta, R. 694, 701, 711
Cristina, F. 752

D

D´andrea, D. 2
Dal Bianco, P. 18
Dapoto, S. 428, 752
Dapozo, G. 248
De Giusti, A. 672
De Giusti, L. 672
De Vincenzi, M. 195
del Rio, N. 11
Desiderio Arana, C. 820
Díaz Lapérgola, M. A. 803
Díaz, A. 81

Díaz, J. F. 439
Diedrichs, A. L. 814, 848
Diedrichs, L. 641
Dieser, P. 652
Domen, M. 662
Dorzán, G. 227
dos Santos, E. 332

E

Elgart, J. F. 146
Encinas, D. 531
Escalante, M. 848
Espíndola, M. C. 248
Etchart, G. R. 67

F

Falcone, N. 797
Farías, A. A. 562, 572, 582
Farías, A. F. 562, 572, 582
Fava, L. A. 439
Fernández Sosa, J. 318
Fernandez, A. 352
Ferrara, D. 803
Ferretti, E. 114
Ferro, F. 388
Fillottrani, P. R. 409
Fisicaro, E. 848
Flores, H. D. 170
Flores, S. R. 451
Fontevila, D. 258

G

Gagliardino, J. J. 146
Ganuza, M. L. 160
Garcia, P. M. 572
García-Díaz, M. E. 88
Garis, A. G. 572
Giacomantone, J. 514
Giménez-Suárez, P. 694
Giorgetti, G. 764
Gómez, S. A. 409
González, A. 604
González, G. 711
González, J. 289
Gonzalez, R. 591, 641
González, S. P. 826
Greiner, C. 248
Guatelli, R. S. 217
Guccione, L. 623

H

Handula, L. 859
Harari, I. 439
Hasperué, W. 18, 97
Hernández, J. 48
Herrera, M. 826
Hidalgo, J. 684

I

Ibañez, E. 672
Ibañez, I. 820
Iglesias, L. 859
Inchausti, P. E. 871
Intas, R. 752
Irrazábal, E. 289
Irusta, D. 848

J

Jimbo Santana, P. 77

K

Kaplan, G. 342
Khün, F. 623
Klenzi, R. 866
Klenzi, R. O. 38
Kreder, M. 871
Kuzman, M. 790, 797

L

Lago, R. 531
Lanzarini, L. 77, 146, 720
Lattuca, A. 814, 848
Lazzurri, G. 623
Lebetti, J. 501
Lechner, M. 541
Ledda, M. 641
Ledesma Vila, S. E. 808
Leguizamón Ovelar, T. M. 88
Leiva La O, R. 591
Lescano, G. 701

Lezcano, A. 289
Lliteras, A. B. 11
López De Luise, D. 662
Ludemann, M. 2
Luna, D. M. 418
Lupi, O. D. 524
Luz Clara, H. 764

M

Maffei, A. L. 58
Maffei, F. 182
Maleh, F. I. 631
Manganelli, S. 615
Manzur, N. 814, 848
Marrero, L. 134
Martin, A. 501
Martín, A. E. 451
Martínez, F. 439
Martinez, M. V. 67
Masanet, M. I. 38, 866
Massa, S. 623
Masson, F. 541
Mazzini, A. 790
Medina, S. 318
Mendieta, D. 541
Mendoza, D. 838
Menini, M. de los A. 694
Mercado, G. 641, 814, 848
Merino, N. 814, 848
Merlino, H. D. 467
Micolini, O. 2
Millan, F. 808
Millán, I. F. 833
Minetti, G. 48, 72, 81
Monfroglio, R. L. 803
Montejano, G. A. 572
Montenegro, A. 308
Montenegro, L. 114
Montezanti, D. 476
Moralejo, R. 641
Morales, M. 531
Moreno, L. 377
Moreyra, M. 541
Moyano, L. G. 124
Muñoz, A. 308
Muñoz, R. 365, 774
Murazzo, M. 457, 501

N

Nahuel, L. 803
Naiouf, M. 318
Navarria, L. 604
Neil, C. 170, 182, 195

O

Oliveros, A. 258
Olsowy, V. 134
Orciari, S. 785
Ortega, F. 790
Ortiz, C. 398
Ortiz, F. 552

P

Pacífico, C. D. 745
Paez, N. 258
Pagano, C. 838
Palma, M. 264
Pandolfi, D. 843
Panessi, W. 398
Panizzi, M. 299, 377, 382, 552, 785
Parlanti, T. S 124
Pascal, A. 27
Pasini, A. 365, 774
Perello, M. G. 106
Perero, L. 352
Pérez Monte, C. 814, 848
Pérez, S. 838
Perez-Monte, C. F. 641
Pesado, P. 134, 365, 428, 774
Petris, R. 248
Piccoli, F. 641
Pinciroli, F. 264
Planas, A. 27
Pollini, J. P. 388
Pollo-Cattaneo, M. F. 631, 820
Ponce-Padilla, C. 701
Puricelli, F. 552
Pytel, P. 631

Q

Quintero, F. P. 418
Quiroga, F. 18

R

Recabarren, F. 866
Riera, G. 342

Rios, G. G. 18
Robles, M. 803
Rodríguez, A. 308
Rodriguez, E. 803
Rodriguez, J. B. 820
Rodríguez, M. 457
Rodriguez, N 352
Rodríguez, N. 457, 501
Romera, L. 289
Romero, G. R. 662
Romero, J. C. 106
Ronchetti, F. 18, 146
Rucci, E. 146, 476, 489
Ruiz, S. 826

> S

Salina, C. 662
Salto, C. 48, 72
Salvatore, L. 524
Sandí-Delgado, J. C. 730
Sanmartino, F. G. 833
Santana, S. 352
Sanz, C. 652, 684, 859
Sapino, M. 114
Sartorio, A. 236
Sattolo, I. 552, 785
Scheffer, M. del V. 833
Scholz, V. 531
Sheffield, N. 541
Silva Pavón, J. M. 11
Silva, S. 48
Simari, G. I. 67
Solana Mendez, M. 790
Sosa-Cabrera, G. 88
Spinelli, A. 623
Stanchi, O. 18
Stauber, F. 27
Suárez, S. L. 476

> T

Taffernaberry, C. 641, 814, 848
Tapparelli, A. G. 871
Taruschio, M. 388, 418
Tellechea, M. 439
Tesone, F. 134
Tetzlaff, T. 58
Teze, J. C. L. 67, 745
Thomas, P. 134, 318, 428
Tittarelli, G. 146
Tobar, M. S. 641
Tobar, S. 814, 848
Turconi, D. 524

> V

Vásquez-Bermúdez, M. 684
Vázquez, P. 299
Vegega, C. 631, 820
Ventre, L. O. 2
Videla, L. 217
Villagra, A. 843
Villarreal, G. L. 672

> Z

Zangara, A. 258, 604, 652, 720
Zangara, M. A. 684
Zapata, M. 814
Zaradnik, I. 524
Zeligueta, L. 264
Zogbe, A. S. 451
Zurita, R. I. 541

ISBN 978-987-9285-51-0



cacic2023.unlu.edu.ar