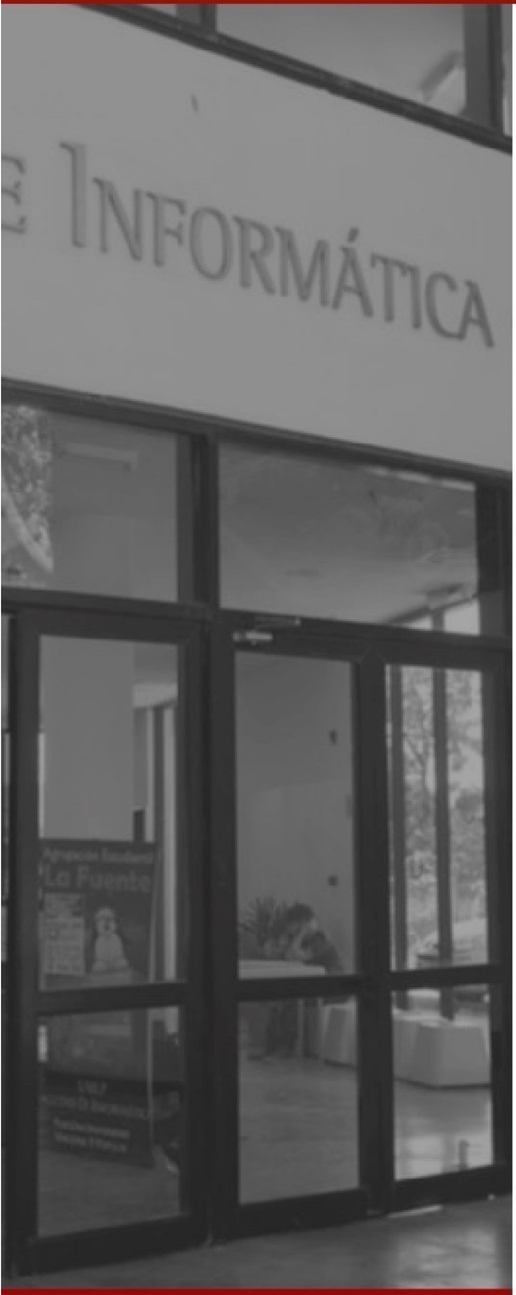
JCC-BD
&ET
2025



13th Conference on Cloud Computing, Big Data & Emerging Topics

(JCC-BD&ET 2025)

HTTPS://JCC.INFO.UNLP.EDU.AR

@CONF_CC_BD_ET

JCC@LIDI.INFO.UNLP.EDU.AR

POSTGRADO
FACULTAD DE INFORMÁTICA

III-LIDI
INSTITUTO DE INVESTIGACIÓN
EN INFORMÁTICA - LIDI

13th Conference on Cloud Computing Conference, Big Data & Emerging Topics

(JCC-BD&ET 2025)

La Plata, Buenos Aires, Argentina.

June 24−26, 2025

13th Conference on Cloud Computing, Big Data & Emerging Topics (JCC-BD&ET 2025) ; Compilación de Marcelo Ricardo Naiouf ... [et al.]. - 1a ed. - La Plata : Universidad Nacional de La Plata, 2025.
Libro digital, PDF

ISBN 978-950-34-2583-1
9 789503 425831

Archivo Digital: descarga y online
ISBN 978-950-34-2583-1

1. Actas de Congresos. 2. Computación. 3. Conferencias. I. Naiouf, Marcelo Ricardo, comp.
CDD 004.01

Preface

Welcome to the proceedings of the 13th Conference on Cloud Computing, Big Data & Emerging Topics (JCC-BD&ET 2025), held in a hybrid modality (both on-site and live online settings were allowed). JCC-BD&ET 2025 was organized by the III-LIDI and the Postgraduate Office, both from School of Computer Science of the National University of La Plata.

Since 2013, this event has been an annual meeting where ideas, projects, scientific results and applications in the cloud computing, big data and other related areas are exchanged and disseminated. The conference focuses on the topics that allow interaction between academia, industry, and other interested parties.

JCC-BD&ET 2025 covered the following topics: high-performance, edge and fog computing; internet of things; modelling and simulation; big and open data; machine and deep learning; smart cities; e-government; human-computer interaction; visualization; and special topics related to emerging technologies. In addition, special activities were also carried out, including 1 plenary lecture and 2 discussion panel.

Special thanks to all the people who contributed to the conference's success: program and organizing committees, authors, reviewers, speakers, and all conference attendees.

June 2025

Marcelo Naiouf
Franco Chichizola
Laura De Giusti
Leandro Libutti

Organization

Program Committee

María José Abásolo	Universidad Nacional de La Plata and CIC, Argentina
José Aguilar	Universidad de Los Andes, Venezuela
Jorge Ardenghi	Universidad Nacional del Sur, Argentina
Javier Balladini	Universidad Nacional del Comahue, Argentina
Oscar Bria	Universidad Nacional de La Plata and INVAP, Argentina
Silvia Castro	Universidad Nacional del Sur, Argentina
Franco Chichizola	Universidad Nacional de La Plata, Argentina
Laura De Giusti	Universidad Nacional de La Plata and CIC, Argentina
Mónica Denham	Universidad Nacional de Río Negro and CONICET, Argentina
Javier Diaz	Universidad Nacional de La Plata, Argentina
Ramón Doallo	Universidade da Coruña, Spain
Marcelo Errecalde	Universidad Nacional de San Luis, Argentina
Elsa Estevez	Universidad Nacional del Sur and CONICET, Argentina
Pablo Ezzatti	Universidad de la República, Uruguay
Aurelio Fernandez Bariviera	Universitat Rovira i Virgili, Spain
Fernando Emmanuel Frati	Universidad Nacional de Chilecito, Argentina
Carlos Garcia Garino	Universidad Nacional de Cuyo, Argentina
Carlos García Sánchez	Universidad Complutense de Madrid, Spain
Adriana Angélica Gaudiani	Universidad Nacional de General Sarmiento, Argentina
Graciela Verónica Gil Costa	Universidad Nacional de San Luis and CONICET, Argentina
Roberto Guerrero	Universidad Nacional de San Luis, Argentina
Waldo Hasperué	Universidad Nacional de La Plata and CIC, Argentina
Francisco Daniel Igual Peña	Universidad Complutense de Madrid, Spain
Tomasz Janowski	Gdansk University of Technology, Poland
Laura Lanzarini	Universidad Nacional de La Plata, Argentina
Guillermo Leguizamón	Universidad Nacional de San Luis, Argentina

Leandro Libutti	Universidad Nacional de La Plata, Argentina
Edimara Luciano	Pontificia Universidade Católica do Rio Grande do Sul, Brazil
Emilio Luque Fadón	Universidad Autónoma de Barcelona, Spain
Mauricio Marín	Universidad de Santiago de Chile, Chile
Luis Marrone	Universidad Nacional de La Plata, Argentina
Marcelo Naiouf	Universidad Nacional de La Plata, Argentina
Katzalin Olcoz Herrero	Universidad Complutense de Madrid, Spain
José Angel Olivas Varela	Universidad de Castilla-La Mancha, Spain
Xoan Pardo	Universidade da Coruña, Spain
Patricia Pesado	Universidad Nacional de La Plata, Argentina
Mario Piattini	Universidad de Castilla-La Mancha, Spain
María Fabiana Piccoli	Universidad Nacional de San Luis, Argentina
Luis Piñuel	Universidad Complutense de Madrid, Spain
Adrian Pousa	Universidad Nacional de La Plata, Argentina
Marcela Printista	Universidad Nacional de San Luis, Argentina
Dolores Isabel Rexachs del Rosario	Universidad Autónoma de Barcelona, Spain
Enzo Rucci	Universidad Nacional de La Plata, Argentina
Nelson Rodríguez	Universidad Nacional de San Juan, Argentina
Juan Carlos Saez Alcaide	Universidad Complutense de Madrid, Spain
Aurora Sánchez	Universidad Católica del Norte, Chile
Victoria Sanz	Universidad Nacional de La Plata, Argentina
Remo Suppi	Universidad Autónoma de Barcelona, Spain
Francisco Tirado Fernández	Universidad Complutense de Madrid, Spain
Juan Touriño Dominguez	Universidade da Coruña, Spain
Gabriela Viale Pereira	Danube University Krems, Austria
Gonzalo Zarza	Globant, Argentina

Additional Reviewers

Hugo Alfonso	
Patricia Bazan	Universidad Nacional de La Plata, Argentina
Alejandra Cechich	Universidad Nacional del Sur, Argentina
Javier Diaz	Universidad Nacional de La Plata, Argentina
César Estrebou	Universidad Nacional de La Plata, Argentina
Diego Encinas	Universidad Nacional de La Plata, Argentina

Alejandro Fernández	Universidad Nacional de La Plata, Argentina
Alberto Fernández	Universidade da Coruña, Spain
Alejandro González	Universidad Nacional de La Plata, Argentina
María Luján Ganuza	Universidad Nacional del Sur, Argentina
Mario Alejandra Garrido	Universidad Nacional de La Plata, Argentina
Marcela Genero	Universidad de Castilla-La Mancha, Spain
Jorge Ierache	Universidad de Buenos Aires. Argentina
Martín Larrea	Universidad Nacional del Sur, Argentina
Lia Molinari	Universidad Nacional de La Plata, Argentina
Francisco Pascual Romero Chicharro	Universidad de Castilla-La Mancha, Spain
Ariel Pasini	Universidad Nacional de La Plata, Argentina
Claudia Pons	Universidad Nacional de La Plata, Argentina
Facundo Quiroga	Universidad Nacional de La Plata, Argentina
Hugo Ramón	Universidad Nacional del Noroeste de la Provincia de Buenos Aires, Argentina
Franco Ronchetti	Universidad Nacional de La Plata, Argentina
Jorge Runco	Universidad Nacional de La Plata, Argentina
Pablo Thomas	Universidad Nacional de La Plata, Argentina
Federico Walas	Universidad Nacional de La Plata, Argentina

Sponsors

III-LIDI

POSTGRADO
FACULTAD DE INFORMÁTICA

Facultad de INFORMÁTICA
UNIVERSIDAD NACIONAL DE LA PLATA

UNIVERSIDAD NACIONAL DE LA PLATA
POR LA CIENCIA Y POR LA PATRIA

RedUNCI

CIC
COMISIÓN DE
INVESTIGACIONES CIENTÍFICAS

CONICET
LA PLATA

Sistemas Nacionales
Computación de Alto
Desempeño

Table of Contents

Full papers

Development and Implementation of an AWS-Based Platform for Automated Prediction of Antibiotic Resistance from Mass Spectrometry Profiles

Moises E. Flores Estay[2,3], Xaviera A. Lopéz Cortés[1,2,3], Felipe Tirado[1,3] José J. I. Bernal Osses[3,4], and José M. Manríquez-Troncoso[1,3]

[1] Departamento de Computación e Industrias, Universidad Católica del Maule, Talca, Chile
[2] Center for Innovation in Applied Engineering (CIIA), Catholic University of Maule, Talca, Chile
[3] Multidisciplinary Intelligent Data Science Lab, Talca, Chile
[4] Facultad de Ingeniería, Universidad de Talca, Talca, Chile

moises.intech@gmail.com,xaviera.lopez.c@gmail.com

Abstract. Antimicrobial resistance (AMR) is a growing threat to global public health, yet few studies have addressed the deployment of intelligent systems for its early prediction in clinical settings. This work presents the development and deployment of a cloud-based web platform that leverages artificial intelligence to predict bacterial resistance using MALDI-TOF mass spectrometry data.

The system was developed using a modular architecture deployed on Amazon Web Services (AWS), combining serverless components and dedicated instances for efficient and scalable operation. A total of 316 clinical isolates of *Escherichia coli* were collected from the Regional Hospital of Talca, Chile, between 2022 and 2023. A benchmarking analysis comparing CatBoost, XGBoost, and LightGBM was conducted to select the most effective boosting algorithm.

The best performance was achieved with Catboost for the Ciprofloxacin case, reaching an AUROC and AUPRC of 0.91. Results for Ceftriaxone were slightly lower, likely due to class imbalance. These outcomes highlight the robustness of boosting models even under real-world data constraints.

The entire platform was deployed on Amazon Web Services (AWS) using a modular serverless architecture, enabling scalability, cost-efficiency, and easy integration into hospital workflows. This study demonstrates the feasibility of integrating AI-powered prediction systems into clinical environments to support timely and data-driven antimicrobial resistance management.

Keywords: Machine learning · Antibiotic resistance · Intelligent systems · AWS Cloud.

1 Introduction

Antimicrobial resistance (AMR) constitutes one of the most global public health threats and has been classified by the World Health Organization as an urgent health crisis of the 21st century [20]. This growing challenge compromises the effectiveness of standard treatments for common bacterial infections, leading to increased morbidity and mortality, as well as significantly elevated healthcare costs [9, 23]. The proliferation of multidrug-resistant bacteria has been fueled by multiple factors, including the indiscriminate use of antibiotics, improper practices in both clinical and community environments, and the absence of diagnostic tools capable of guiding timely and accurate treatment decisions [15]. In this context, the development of technological solutions that can anticipate resistance patterns and support early-stage antibiotic therapy selection has become a global priority.

Mass spectrometry, particularly the MALDI-TOF (Matrix-Assisted Laser Desorption/Ionization Time-of-Flight) technique, has recently emerged as a transformative tool in the rapid identification of microorganisms [8]. Its capacity to produce protein spectra in a matter of seconds has revolutionized microbiological diagnostics in clinical laboratories. Nevertheless, the full potential of this technology is unlocked when it is combined with machine learning (ML) techniques. These computational models are capable of processing large volumes of spectral data and identifying subtle patterns associated with bacterial resistance mechanisms that are not evident using traditional diagnostic methods [24]. Recent studies have demonstrated that this synergy between mass spectrometry (MS) and machine learning not only improves diagnostic precision but also significantly reduces the time required to determine resistance profiles [4, 12, 16–18].

The integration of intelligent systems into hospital workflows further amplifies these benefits by automating key clinical tasks, such as sample analysis, antibiotic selection, and early detection of resistant strains [22]. These systems also enable centralized access to relevant clinical information, enhance treatment traceability, and support evidence-based decision-making [13]. Collectively, such improvements contribute to alleviating the workload of healthcare professionals while optimizing the use of institutional resources.

In this context, we present the design, implementation, and deployment of a cloud-based web platform that leverages artificial intelligence to predict antimicrobial resistance. This computational solution, fully deployed on cloud infrastructure, enables the automated processing of bacterial samples and delivers real-time resistance predictions based on supervised learning models trained on clinical data. This study focused specifically on the bacterial species *Escherichia coli* and the antibiotics ciprofloxacin and ceftriaxone, which are commonly used in clinical practice. To select the most accurate predictive model, a benchmarking analysis of state-of-the-art gradient boosting algorithms, specifically XGBoost, LightGBM, and CatBoost was performed. The platform architecture was implemented in Amazon Web Services (AWS) and follows a modular, secure, and scalable design, incorporating features such as role-based access control, asynchronous task processing, and extensible predictive components tailored for practical use in healthcare environments.

2 State of the Art

The development of digital platforms to support clinical decision-making through machine learning (ML) has advanced significantly in recent years. In the context of antimicrobial resistance (AMR), multiple approaches have emerged that incorporate modern technologies such as cloud computing, serverless inference, and web-based visualization.

One of the pioneering tools in this domain is ResFinder [25], a web service designed to detect acquired AMR genes from whole-genome sequencing data. While its approach is not predictive, it stands out for providing an accessible interface to professionals without requiring bioinformatics expertise.

Another notable system is ResistanceOpen [19], a web-based platform that aggregates and visualizes regional AMR patterns using publicly available data sources. Although it lacks predictive capabilities, it shares with Mindslab the vision of providing a user-friendly and clinically useful web interface.

Regarding the use of ML with mass spectrometry data, recent studies have demonstrated the potential of combining MALDI-TOF MS with predictive models. For instance, Oviaño et al. [21] accelerated the detection of AMR in blood cultures through an optimized MALDI-TOF protocol, but without integration into a web interface or cloud-native environment.

From a technological standpoint, the Serverless on FHIR architecture [10] offers a modular, cloud-native deployment of ML models using AWS Lambda and integration with FHIR-based clinical data systems. This architecture enables automatic scaling, cost reduction, and compliance with security standards—all characteristics adopted by Mindslab.

Generic bioinformatics platforms such as Bio-OS [11] and gcMeta [6] have shown the benefits of cloud-based, reusable ecosystems for biological data analysis. However, they lack clinical specialization and real-time prediction features.

Despite these advances, no previous work integrates real-time AMR prediction from MALDI-TOF spectra with a serverless deployment model and an intuitive web-based user interface. Therefore, the present platform, **Mindslab**, represents a novel contribution at the intersection of predictive analytics, microbiology, and scalable software architecture.

3 Materials and Methods

3.1 Data

In the present study, mass spectrometry data consisting of a collection of MALDI-TOF mass spectrometry data from *Escherichia coli* bacteria gathered at the Regional Hospital of Talca, Chile, from September 2022 to July 2023 was used.

These samples originated from various clinical sources (urine, respiratory secretions, wound secretions, surgical wounds, bone fragments, abscesses, blood cultures, sterile fluids, among others), obtained from patients within the healthcare network. Each sample was cultured on commercial media such as Columbia Agar and MacConkey Agar (VALTEK, Santiago, Chile) to promote bacterial growth and incubated at 37 °C for 24 hours. Bacterial colonies were collected and subjected to species identification using the VITEK®MS mass spectrometer (Biomerieux, Paris, France).

For susceptibility testing, the Kirby-Bauer disk diffusion method was employed using Muller Hinton II plates (VALTEK, Santiago, Chile). Each bacterial colony was adjusted to a concentration of 0.5 McFarland standard and exposed to various antibiotics. The results were visually assessed by measuring inhibition zone diameters for each antibiotic following the guidelines provided in the "Performance Standards for Antimicrobial Susceptibility Testing" CLSI M100 ED33:2023.

Preproccessing To prepare the dataset suitable for machine learning algorithms, preprocessing of the mass spectra obtained from the VITEK®MS equipment was initially required [17]. Briefly, the instrument performs baseline removal, smoothing, and peak detection, resulting in a summarized spectrum of approximately 200 m/z peaks distributed between 2,000 Da and 12,000 Da. To obtain fixed-length vectors, mass spectra were discretized using a binning process, which involves grouping measured mass values into discrete ranges or "bins," where the representative value corresponds to the mean intensity within each bin. Binning was applied within the range of 2,000 to 10,000 Da with a bin size of 5 Da, ensuring an appropriate distribution of mass peaks. In addition to spectral processing, each spectrum needed to be linked to its corresponding antibiotic resistance labels derived from antimicrobial susceptibility tests using a unique numerical code associating the laboratory report with the spectrum provided by the VITEK®MS instrument. Table 1 shows the number of available samples for the current case study, focusing on the bacterium *Escherichia coli*, selected for its clinical relevance, the quantity of available samples, and its inclusion within critical research interest groups as indicated by the World Health Organization [20].

Table 1. Class distribution (resistant vs. susceptible) of *E. coli* samples for each antibiotic under study: Ciprofloxacin and Ceftriaxone.

Bacteria	Antibiotic	Resistant	Susceptible
Escherichia coli	Ciprofloxacin	107	145
	Ceftriaxone	46	162

3.2 Machine learning models and performance metrics

For the development of the models used in the AWS platform, a benchmarking analysis among the following gradient boosting algorithms, XGBoost, LightGBM, and CatBoost, was implemented. In this new study, we decided to explore alternative gradient boosting algorithms based on the findings of our previous work [17], where CatBoost emerged as the most effective predictor for antimicrobial resistance. Motivated by these results, we extended the analysis to include a comparative evaluation of other state-of-the-art boosting algorithms such as, XGBoost and LightGBM with the aim of validating or potentially improving upon the performance achieved with CatBoost.

The performance comparison among XGBoost, LightGBM, and CatBoost was conducted using cross-validation and evaluated through key metrics, including the Area Under the Precision-Recall Curve (AUPRC), training time, and model interpretability. To optimize the model configuration, 10-fold cross-validation was used within a Bayesian hyperparameter search. During this hyperparameter search, the AUPRC was optimized. This metric is calculated based on precision and recall, focusing on the positive class, which makes it the ideal metric for accurately evaluating binary classification models in imbalanced data environments. The optimized parameters were 'iterations', 'depth', 'l2_leaf_reg', and 'learning_rate'. Additionally, to further improve model accuracy, the recursive feature elimination with cross-validation (RFECV) method was implemented.

This approach enabled the automatic selection of the optimal set of relevant features for each of the studied cases. The RFECV process involved iteratively assessing the relevance of features using a base model, eliminating the least informative ones, and validating the model's performance through stratified 10-fold cross-validation. This procedure was integrated as a preliminary step prior to the hyperparameter search, ensuring that the optimal configurations were applied only to the features selected by RFECV.

Finally, the resulting models were saved in .pkl format, which is characterized by its easy integration with popular libraries (i.e. Scikit-learn and TensorFlow), fast loading via pickle or joblib, and compatibility with complex structures such as pipelines and preprocessors. Additionally, it allows compact storage in binary format, optimizing memory usage and reducing transfer time in web-based systems.

3.3 AWS Architecture

The platform stack leverages serverless, cloud-native services provided by Amazon Web Services (AWS). This section first describes the infrastructure (Figure 1 and Table 2) and then explains how the design achieves modularity, security, and scalability.

Figure 1 illustrates the cloud architecture of the MindsLab platform, built on AWS services. The architecture follows a three-layered structure that organizes its functionality: the presentation layer delivers a single-page application and handles user interactions; the API and processing layer orchestrates synchronous requests and asynchronous processing through API Gateway and Lambda functions; and the data and model layer manages the storage of raw data, metadata, and clinical metrics, as well as the execution of inference models. This layered design enables scalable deployment, modular development, and efficient system maintenance.

Table 2 summarizes the AWS services integrated into the platform and their specific roles within the system architecture. By leveraging these services, the platform ensures high availability, scalability, and operational efficiency in executing core tasks, including machine learning inference, clinical data management, user authentication, and system monitoring.

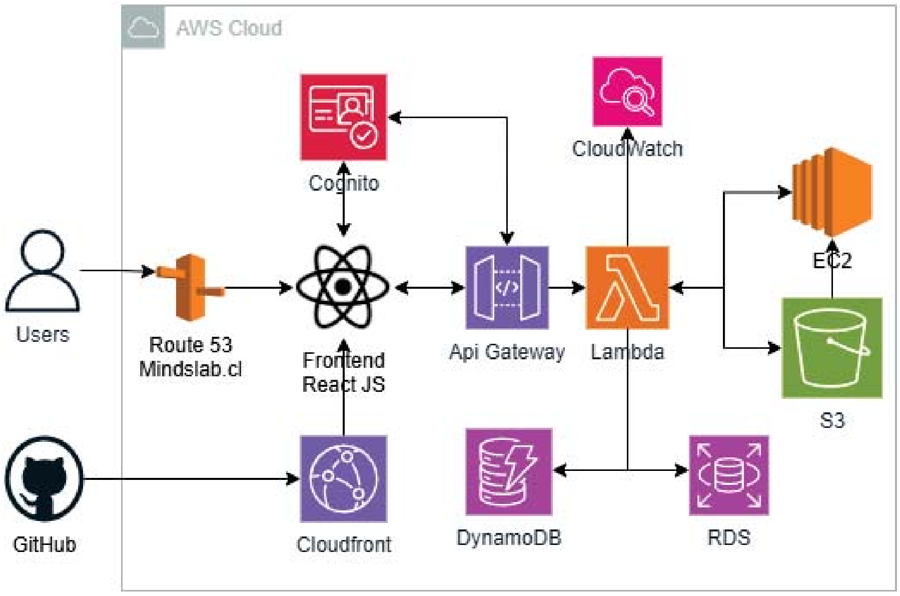


Figure 1. High-level cloud architecture of the MindsLab platform. The architecture follows a three-layer structure: (L1) the presentation layer delivers a single-page React application via Amazon CloudFront; (L2) the API and processing layer uses Amazon API Gateway and AWS Lambda to coordinate synchronous requests and asynchronous tasks; and (L3) the data and models layer stores spectra and models in Amazon S3, maintains per-sample metadata in Amazon DynamoDB, and aggregates clinical metrics in Amazon RDS (MySQL).

Table 2. AWS services and their responsibility within the platform.

Service	Responsibility
Amazon API Gateway	HTTPS / WebSocket entry points; throttling; auth hooks
AWS Lambda	Business logic per endpoint (13 functions)
Amazon SQS	Queue buffering inference jobs
Amazon EC2 (GPU)	Batch inference on CatBoost / XGBoost / LightGBM models
Amazon DynamoDB	NoSQL metadata store keyed by `sampleId`
Amazon RDS (MySQL)	Aggregated clinical reporting data
Amazon S3	Raw spectra and versioned model artefacts
Amazon CloudFront	Content Delivery Network for the SPA bundle
Amazon Cognito	User authentication and JWT issuance (RBAC)
Amazon CloudWatch	Centralised logs, alarms, dashboards
Amazon Route 53	Authoritative DNS for `https://mindslab.cl`

The resulting architecture exhibits three key properties that are essential for a robust and maintainable cloud-based system. First, the platform achieves modularity by encapsulating business logic into thirteen independent AWS Lambda functions while storing static assets and machine learning models in Amazon S3. This decoupling enables the addition or update of new models without requiring modifications to the core application code. Second, it enforces security through Amazon Cognito [1], which provides role-based access control (RBAC) and manages user authentication and authorization. All data are encrypted in transit using TLS 1.2 and at rest using SSE-S3 and AES-256 encryption standards, in alignment with AWS's Security Pillar best practices. Finally, the platform supports scalability through the automatic resource provisioning capabilities of API Gateway and AWS Lambda [3].

Benefits of Serverless Architecture The implementation of a serverless architecture in the proposed platform delivers clear advantages in terms of scalability, maintainability, and cost effectiveness. By abstracting away the management of physical or virtual servers, serverless computing (through services such as AWS Lambda, API Gateway, and DynamoDB.) enables automatic resource allocation based on demand [2, 14]. This model ensures high availability and fault tolerance by design without requiring manual intervention or overprovisioning of infrastructure. The overall design aligns with the principles of the AWS Well Architected Framework, particularly the pillars of operational excellence, performance efficiency, and cost optimization. Among the most relevant benefits are the following:

- Automatic scaling: Backend services, such as Lambda functions, dynamically scale in response to workload fluctuations, maintaining consistent performance regardless of traffic peaks or idle periods.
- Reduced operational complexity: The development team can concentrate on building business logic and integrating machine learning workflows rather than maintaining and configuring servers.

- Optimized costs: The billing model charges only for actual compute time and API usage, with no costs associated with idle resources, making this approach especially suitable for workloads with variable or unpredictable demand.
- Faster deployment cycles: The modular and function oriented architecture allows isolated updates and seamless integration into continuous integration and deployment (CI/CD) pipelines, thereby reducing regression risks and shortening development iterations

In clinical settings, where usage patterns can be highly irregular and performance demands are stringent, a serverless infrastructure offers a sustainable, elastic, and agile foundation for deploying AI-based solutions at scale.

Front-End Layer The front end, built with React, operates as a single-page application to improve responsiveness and eliminate full-page reloads. To enhance usability and ensure an intuitive user experience, we introduced several design improvements focused on clarity and guidance.

Redesigned the data upload form as a sequential, step-by-step interface. Each step is clearly numbered, helping users understand the required actions and follow the correct order. The form includes real-time validation that highlights missing or incorrect fields as the user progresses, reducing the likelihood of submission errors. Required fields are visually marked, and tooltips provide immediate, contextual explanations. Once the system generates a prediction, the interface displays results using color-coded tags and short, informative messages to make the output easy to interpret. These changes streamline the data entry process, enhance user confidence, and facilitate efficient interaction, which is especially valuable in clinical workflows where accuracy and speed are crucial.

4 Results

The following section presents the results obtained by deploying the web platform and applying it to real-world data from *Escherichia coli* samples acquired through MALDI-TOF mass spectrometry. We enhanced the platform by integrating a gradient boosting model selected through a rigorous benchmarking process. This integration significantly improved the accuracy and robustness of antimicrobial resistance detection, demonstrating the practical value of combining advanced machine learning methods with clinical data.

4.1 Workflow for Antibiotic Resistance Prediction

Figure 2 illustrates the operational workflow of the proposed system, from the collection of clinical samples to the generation of predictions through the platform. The process begins with the acquisition of biological samples (blood, urine, tissue, among others), which are analyzed using mass spectrometry through the

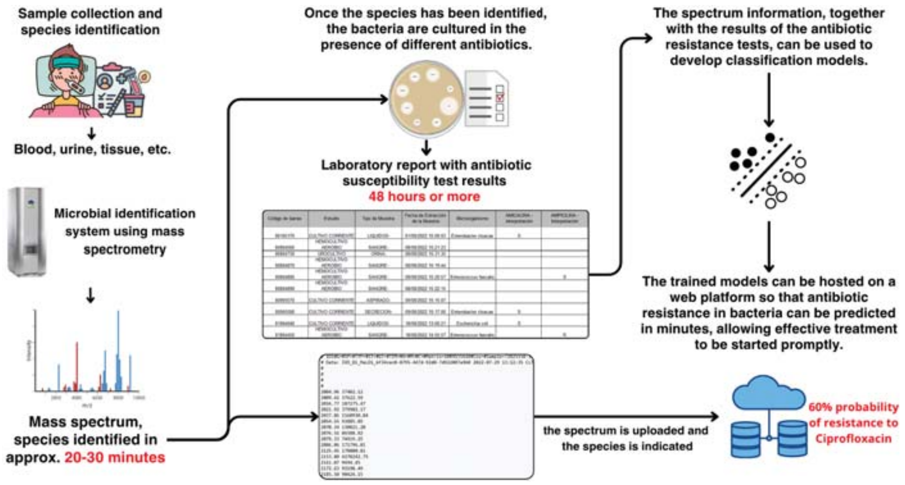

Sample collection and species identification
Blood, urine, tissue, etc.
Microbial identification system using mass spectrometry
Mass spectrum, species identified in approx. 20-30 minutes
Once the species has been identified, the bacteria are cultured in the presence of different antibiotics.
Laboratory report with antibiotic susceptibility test results
48 hours or more
The spectrum information, together with the results of the antibiotic resistance tests, can be used to develop classification models.
The trained models can be hosted on a web platform so that antibiotic resistance in bacteria can be predicted in minutes, allowing effective treatment to be started promptly.
the spectrum is uploaded and the species is indicated
60% probability of resistance to Ciprofloxacin

Figure 2. Workflow for antibiotic resistance prediction.

VITEK MS system, enabling the identification of the bacterial species within 20 to 30 minutes.

Traditionally, a bacterial culture in the presence of various antibiotics is required to determine the susceptibility profile, a procedure that can take at least 48 hours. In contrast, the platform system offers a solution based on classification models trained with spectral data and historical antibiogram results, allowing resistance probabilities to be predicted within minutes.

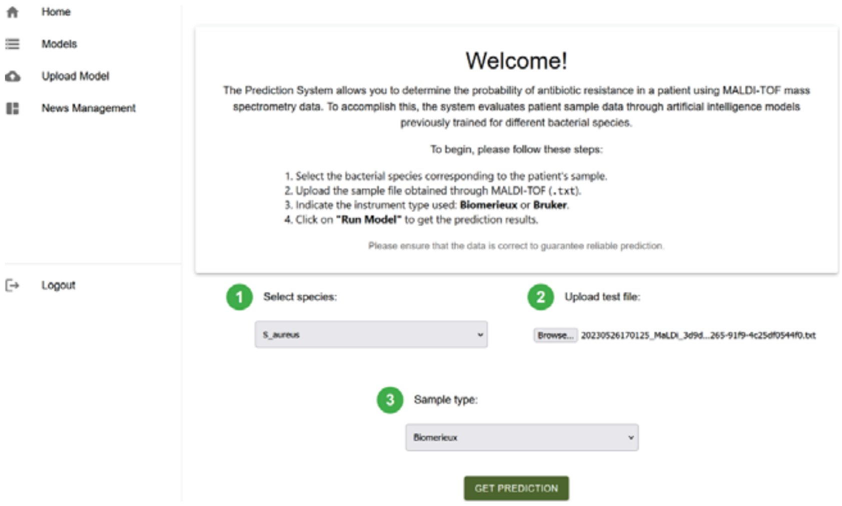
Once deployed on the web platform, these models enable healthcare personnel to upload the spectrum and select the identified species to obtain a real-time probabilistic estimation of resistance to various antibiotics.

4.2 Sample Upload Interface

Figure 3 shows the platform's web interface, which allows healthcare professionals and laboratory staff to upload new bacterial samples. Users submit a MALDI-TOF mass spectrometry file along with a unique identifier, ensuring sample traceability and enabling direct linkage between prediction results and clinical records. The interface supports real-time interaction with the prediction engine while maintaining data structure and preserving patient context.

4.3 Models

Table 3 shows the results obtained for each case study, reporting the mean and standard deviation after performing a 10-fold cross-validation. As part of the



Welcome!

The Prediction System allows you to determine the probability of antibiotic resistance in a patient using MALDI-TOF mass spectrometry data. To accomplish this, the system evaluates patient sample data through artificial intelligence models previously trained for different bacterial species.

To begin, please follow these steps:

1. Select the bacterial species corresponding to the patient's sample.
2. Upload the sample file obtained through MALDI-TOF (`.txt`).
3. Indicate the instrument type used: **Biomerieux** or **Bruker**.
4. Click on **"Run Model"** to get the prediction results.

Please ensure that the data is correct to guarantee reliable prediction.

1 Select species: S_aureus

2 Upload test file: Browse... 20230526170125_MaLDi_3d9d...265-91f9-4c25df0544f0.txt

3 Sample type: Biomerieux

Figure 3. Main web interface for submitting a prediction job.

platform enhancement, we integrated and benchmarked three gradient boosting algorithms: CatBoost, XGBoost, and LightGBM, selected for their strong performance in tabular data classification tasks.

For the *E. coli*-Ciprofloxacin case, CatBoost achieved outstanding performance with AUROC and AUPRC values of 0.91, balanced accuracy of 0.81, and an F1-score of 0.78. In comparison, XGBoost yielded lower but still acceptable results, with 0.81 in both AUROC and AUPRC, 0.72 in balanced accuracy, and 0.64 in F1-score. LightGBM showed performance of 0.82 in AUROC and 0.80 in AUPRC, with a balanced accuracy of 0.72 and F1-score of 0.66.

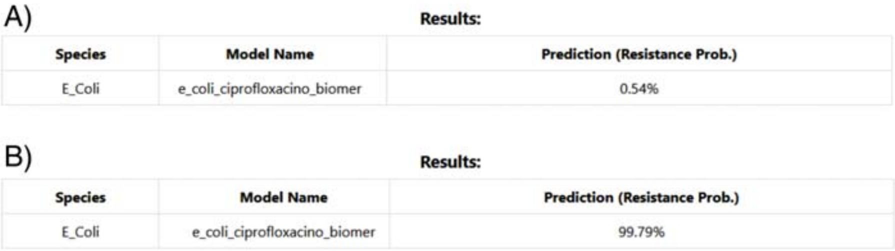
In the *E. coli*-Ceftriaxone case, all models performed less favorably, likely due to class imbalance. CatBoost obtained an AUROC of 0.78 and an AUPRC of 0.71, while XGBoost and LightGBM reported values of 0.69 and 0.52, respectively, across the evaluation metrics. These results highlight the importance of model selection and the potential benefits of advanced boosting techniques for antimicrobial resistance prediction.

4.4 Antimicrobial Resistance Prediction

Once users upload a sample, the system processes it automatically using CatBoost, the best-performing machine learning model identified during benchmarking. Figure 4 presents two screenshots of the platform that illustrate prediction outcomes: one sample is classified as resistant to ciprofloxacin, while the other is identified as susceptible. These examples demonstrate the platform's ability to generate timely and interpretable predictions, supporting informed clinical decision-making.

Table 3. Performance metrics for each case study using CatBoost, XGBoost, and LightGBM algorithms.

Antibiotic	Algorithm	AUROC	AUPRC	B. Acc	F1-Score
Ciprofloxacin	Catboost	0.91±0.07	0.91±0.06	0.81±0.08	0.78±0.08
	XGBoost	0.84±0.03	0.84±0.02	0.72±0.03	0.64±0.08
	LightGBM	0.82±0.08	0.80±0.07	0.72±0.08	0.66±0.12
Ceftriaxone	Catboost	0.78±0.05	0.71±0.06	0.78±0.05	0.73±0.07
	XGBoost	0.69±0.04	0.61±0.05	0.67±0.07	0.60±0.04
	LightGBM	0.69±0.07	0.52±0.09	0.57±0.05	0.23±0.16



A)

Results:

Species	Model Name	Prediction (Resistance Prob.)
E_Coli	e_coli_ciprofloxacino_biomer	0.54%

B)

Results:

Species	Model Name	Prediction (Resistance Prob.)
E_Coli	e_coli_ciprofloxacino_biomer	99.79%

Figure 4. Example of a prediction result generated by the platform: A) Sample classified as resistant to Ciprofloxacin. B) Sample classified as susceptible to Ciprofloxacin.

4.5 User-facing web platform, usability and stress test

The MindsLab platform combines architectural robustness in the cloud with functional capabilities as a web-based clinical tool. We developed the front end using React and deployed it globally through Amazon CloudFront, ensuring high availability and low latency access. This interface enables clinicians and researchers to interact directly with predictive models from a standard web browser, eliminating the need for additional software or configuration.

Main Functionalities The system is divided into three primary sections:

1. **Prediction Interface (Home)**: Users can upload a patient sample in `.txt` format obtained from MALDI-TOF instruments, specify the bacterial species and instrument type (Biomerieux or Bruker), and run the selected model with a single click. Predictions are returned in seconds, and the UI provides step-by-step guidance to ensure correctness of input (Figure 3).
2. **Model Upload Section**: Researchers can add new models via a dedicated interface. Each upload requires three items: the model file (`.pkl`), a JSON-formatted feature list, and metadata such as bacterial species and sample type. Once uploaded, the model becomes immediately available for use in the prediction interface (Figure 1 in Supplementary material).
3. **Model Management**: A table view allows users to manage existing models, including editing metadata or deleting unused models. This encourages continuous improvement and controlled experimentation with new predictive techniques (Figure 2 in Supplementary material).

User Experience Observations The interface was designed to be intuitive and action-oriented. Step numbering in the prediction form, combined with drop-down menus and input validation, helps users avoid common errors. This design enables non-technical users to run machine learning models within clinical workflows without requiring code or interaction with cloud infrastructure.

Key benefits include a short learning curve for first-time users and immediate feedback after each interaction. By abstracting the underlying technical complexity and exposing powerful AI models through a streamlined interface, the MindsLab platform offers a practical example of *AI-as-a-service for clinical microbiology*

Usability Evaluation with SUS To empirically assess the usability of the Mindslab interface, a System Usability Scale (SUS) questionnaire was presented to users immediately after completing a prediction. The SUS is a standardized 10-item survey designed to measure perceived ease of use, complexity, and confidence [5].

A total of 15 participants, including microbiologists and medical technologists, completed the questionnaire. Figure 5 presents the individual scores recorded in the system. The supplementary material includes the survey integrated into the platform (Figure 3 in supplementary material).

The responses yielded an average SUS score of 79.67, with values ranging from 60 to 92. According to industry benchmarks, this score corresponds to a *Good to Excellent* usability rating, confirming that users—regardless of technical background—were able to interact with the platform effectively and confidently.

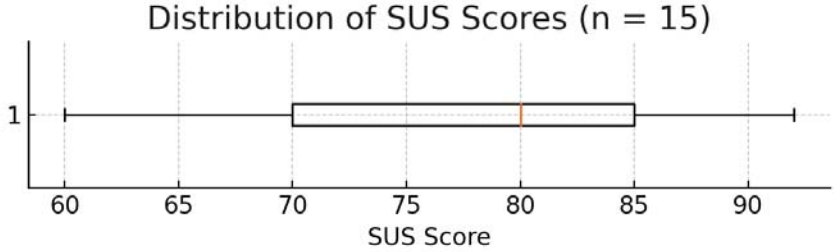


Figure 5. Boxplot of SUS score distribution for the 15 participants (scores retrieved from DynamoDB).

Stress Tests To evaluate the robustness, responsiveness, and scalability of the AWS-based platform, we conducted an updated stress testing campaign using a custom Python script built with the `aiohttp` library. This script simulated high-concurrency scenarios by launching multiple asynchronous workers that simultaneously targeted the `/stressModel` endpoint.

We executed four test scenarios with increasing levels of concurrency: 100, 200, 300, and 500 parallel workers. Each worker continuously sent inference requests until reaching the predefined request count for its scenario. During each test, we recorded key performance metrics, including minimum, maximum, mean, median (50th percentile), 90th percentile, and 99th percentile latencies. These results enabled us to evaluate the platform's performance under progressively higher loads and to identify potential bottlenecks in real-time inference workflows.

The results are summarized below:

- **100 workers**: mean latency of 4.59 s (min 2.27 s, max 5.12 s)
- **200 workers**: mean latency of 7.19 s (min 2.56 s, max 9.05 s)
- **300 workers**: mean latency of 9.39 s (min 3.53 s, max 12.98 s)
- **500 workers**: mean latency of 12.88 s (min 2.61 s, max 20.25 s)

Importantly, all requests in all scenarios completed successfully without any failures or dropped connections, confirming the reliability of the backend services under sustained high concurrency.

Figure 6 illustrates the latency distributions and throughput across the different concurrency levels. As expected, latency increased as the number of simultaneous requests grew; however, the system maintained functional stability and returned valid predictions in all cases.

These findings validate that the Mindslab platform's serverless and container-based architecture can withstand short-term peak usage, supporting real-time clinical workloads with predictable performance. They also highlight potential optimization opportunities for scaling inference to reduce response times further as user demand increases

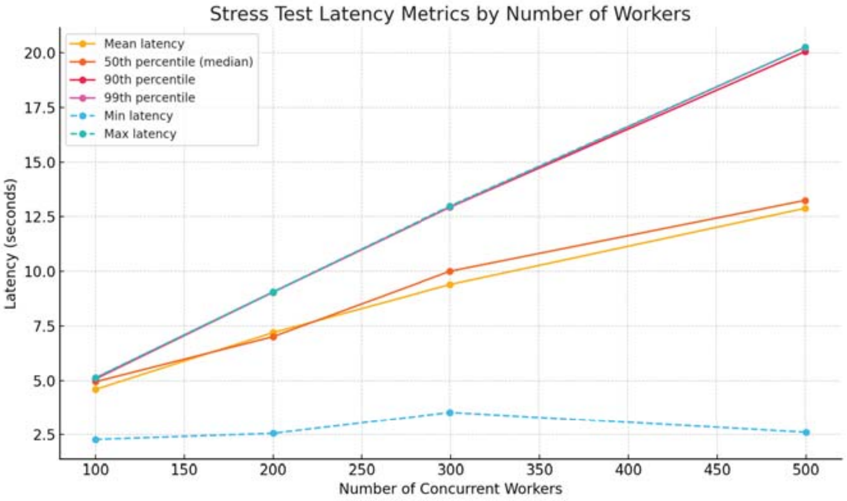


Figure 6. Stress test latency metrics for 100, 200, 300, and 500 concurrent workers.

Low Costs and Economic Efficiency One of the most important benefits of using serverless and on-demand AWS services is the significant reduction in operational costs. By eliminating the need to provision and maintain always-on infrastructure, the system incurs costs strictly based on actual usage, aligning resource consumption with demand. This pay-per-use model leads to significant savings in operational expenditure and promotes more sustainable budgeting, especially in environments with variable or unpredictable workloads.

Between January and May 2025, the platform incurred a total cost of USD 88.46, averaging USD 17.69 per month. This includes all core AWS services, such as model storage, inference, content delivery, and authentication. The predictable monthly pattern confirms the economic feasibility of the serverless model even under real clinical usage conditions. This provides continuous access to predictive models and ensures the permanent availability of the `mindslab.cl` domain. Moreover, as the number of users increases, the incremental cost associated with additional AWS Lambda invocations remains proportionally low relative to the total cost, allowing the platform to scale efficiently with minimal financial impact.

5 Discussion

This work explored the development of a cloud-based web platform to automate the prediction of antibiotic resistance using MALDI-TOF mass spectrometry data and machine learning techniques. Our results highlight the strong performance and robustness of the CatBoost algorithm, which emerged as the best predictor in our previous study [17]. The results obtained with the CatBoost algorithm demonstrate its strong performance and suitability for this task, even when working with a relatively small and imbalanced dataset. These findings align with prior studies that have highlighted the effectiveness of gradient boosting models when applied to heterogeneous biomedical data such as mass spectra [7].

Furthermore, the current study extended the analysis by benchmarking CatBoost against other other gradient boosting algorithms: XGBoost and LightGBM. The comparative evaluation, based on cross-validated metrics such as AUROC and AUPRC, training time, and model interpretability, confirmed CatBoost's superior predictive accuracy for the *Escherichia coli* resistance profiles to ciprofloxacin and ceftriaxone. While XGBoost and LightGBM showed competitive results, CatBoost consistently outperformed them, justifying its selection for integration within the deployed platform.

From a systems development perspective, the implementation of the web platform using a modular serverless architecture on AWS proved effective. This approach enables automated processing, rapid deployment, horizontal scalability, and secure access control, all while maintaining low operational costs. The integration of AWS services such as Lambda, API Gateway, Cognito, and S3 allowed seamless management of machine learning inference workflows, and the use of EC2 instances to handle computationally intensive real-time spectral analysis met the platform's performance demands.

Despite these advances, several avenues remain for improvement and future work. Expanding the model scope to cover additional bacterial species and a broader range of antibiotics would enhance clinical applicability. Besides, incorporating explainability techniques (e.g., SHAP values) could increase trust and interpretability for healthcare professionals. Moreover, integrating the platform with hospital information systems (HIS) would facilitate real-time clinical decision support and streamline laboratory workflows.

Finally, this study has some limitations, including the relatively small and imbalanced dataset, which restricts the generalizability of the models across different pathogens and geographical settings. Additionally, although the platform performs well under controlled conditions, clinical validation in routine hospital environments is necessary to confirm its practical utility. Future work should focus on enlarging and diversifying the dataset, improving model robustness, and conducting prospective validation studies in diverse healthcare settings to ensure wider adoption.

6 Conclusion

This work presents the design, development, and evaluation of a cloud-based web platform for predicting antimicrobial resistance using MALDI-TOF mass spectrometry data and machine learning models. Through a comprehensive methodology that included data preprocessing, algorithm benchmarking, and cloud deployment, the platform demonstrates both technical feasibility and practical relevance in clinical microbiology.

Among the evaluated models, CatBoost achieved the best performance metrics in predicting resistance to Ciprofloxacin and Ceftriaxone in *Escherichia coli*, outperforming XGBoost and LightGBM in terms of AUROC and AUPRC. This reinforces the suitability of gradient boosting algorithms for processing complex biomedical data such as mass spectra.

The deployment of the system on AWS using a serverless architecture proved advantageous in terms of scalability, security, and cost-efficiency. Moreover, stress testing confirmed the platform's ability to handle sample uploads and predictions with low latency and controlled operational costs, ensuring its viability for real-world clinical environments.

In summary, this platform not only automates antimicrobial resistance prediction with high accuracy but also provides a scalable and user-friendly tool to support clinical decision-making. Future work will focus on expanding bacterial coverage, integrating the platform into hospital systems, and validating its clinical impact in prospective studies.

Acknowledgements

The authors gratefully acknowledge financial support from Research Project ANID FONDECYT Iniciación en Investigación No. 11220897.

Bibliography

[1] Amazon Web Services: Security pillar – aws well-architected framework. Tech. rep., Amazon Web Services (2024), `https://docs.aws.amazon.com/wellarchitected/latest/security-pillar/security-pillar.pdf`, accessed: 2025-06-05

[2] Amazon Web Services: Serverless applications lens – aws well-architected framework. Tech. rep., Amazon Web Services (2024), `https://docs.aws.amazon.com/wellarchitected/latest/serverless-applications-lens/serverless-applications-lens.html`, accessed: 2025-06-05

[3] Amazon Web Services: AWS Lambda Developer Guide (2025), `https://docs.aws.amazon.com/lambda/latest/dg/welcome.html`, accessed: 2025-06-05

[4] Astudillo, C.A., López-Cortés, X.A., Ocque, E., Manríquez-Troncoso, J.M.: Multi-label classification to predict antibiotic resistance from raw clinical maldi-tof mass spectrometry data. Scientific Reports **14**(1), 31283 (2024)

[5] Brooke, J.: SUS: A 'Quick and Dirty' Usability Scale, pp. 189–194. Taylor & Francis (1996)

[6] Cheng, L., Qi, Q., Zhang, B., Liu, Q., Liu, X., Qin, N., Zhang, W., Zhu, X., Ren, Y., Wang, F., et al.: gcmeta: a global catalogue of metagenomics platform to support the archiving, standardization and analysis of microbiome data. Nucleic Acids Research **45**(D1), D611–D618 (2017)

[7] Chung, C.R., Wang, H.Y., Yao, C.H., Wu, L.C., Lu, J.J., Horng, J.T., Lee, T.Y.: Data-driven two-stage framework for identification and characterization of different antibiotic-resistant escherichia coli isolates based on mass spectrometry data. Microbiology spectrum **11**(3), e03479–22 (2023)

[8] Clark, A.E., Kaleta, E.J., Arora, A., Wolk, D.M.: Matrix-assisted laser desorption ionization–time of flight mass spectrometry: a fundamental shift in the routine practice of clinical microbiology. Clinical Microbiology Reviews **26**(3), 547–603 (2013)

[9] for Disease Control, C., (CDC), P.: Antibiotic resistance threats in the united states, 2019 (2019), available at: `https://www.cdc.gov/drugresistance/pdf/threats-report/2019-ar-threats-report-508.pdf`

[10] Eapen, B., Tiwari, A., de Mello, R., Nair, R., Mohamed, K., D'Souza, M., Yaghoobi, D., Ong, M.Y., Uddin, M., Wilson, P.: Serverless on fhir: Deploying portable machine learning on the cloud to support clinical decision making. Applied Clinical Informatics **12**(2), 367–376 (2021)

[11] Gavrikov, D., Chen, J., Peng, Y., Liu, X., Wu, T., Wang, X., et al.: Bio-os: A bio-medical big data operating system for omics pipelines in the cloud. Genes **14**(2), 301 (2023)

[12] Greco, G., Gholami, H., Geschwindner, S., et al.: Maldi-tof mass spectrometry and machine learning for the diagnosis of antimicrobial resistance: a review. Expert Review of Proteomics **18**(4), 319–330 (2021)

[13] Jiang, F., Jiang, Y., Zhi, H., et al.: Artificial intelligence in healthcare: past, present and future. Stroke and Vascular Neurology **2**(4), 230–243 (2017)

[14] Jonas, E., Schleier-Smith, J., Sreekanti, V., Tsai, C.C., Khandelwal, A., Pu, Q., Shankar, V., Carreira, J., Krauth, K., Yadwadkar, N., Gonzalez, J.E., Popa, R.A., Stoica, I., Patterson, D.A.: Cloud programming simplified: A berkeley view on serverless computing. Tech. Rep. UCB/EECS-2019-3, University of California, Berkeley (2019), `https://arxiv.org/abs/1902.03383`, accessed: 2025-06-05

[15] Laxminarayan, R., Duse, A., Wattal, C., et al.: Antibiotic resistance—the need for global solutions. The Lancet Infectious Diseases **13**(12), 1057–1098 (2013)

[16] López-Cortés, X.A., Manríquez-Troncoso, J.M., Hernández-García, R., Peralta, D.: Msdeepamr: antimicrobial resistance prediction based on deep neural networks and transfer learning. Frontiers in Microbiology **15**, 1361795 (2024)

[17] López-Cortés, X.A., Manríquez-Troncoso, J.M., Sepúlveda, A.Y., Soto, P.S.: Integrating machine learning with maldi-tof mass spectrometry for rapid and accurate antimicrobial resistance detection in clinical pathogens. International Journal of Molecular Sciences **26**(3), 1140 (2025)

[18] Macaya Mejias, V., Zabala-Blanco, D., López-Cortés, X.A., Tirado, F., Manríquez-Troncoso, J.M., Ahumada-García, R.: Predicting bacterial antibiotic resistance using maldi-tof mass spectrometry databases with elm applications. Journal of Computer Science & Technology **24** (2024)

[19] MacFadden, D.R., Fisman, D.N., Andre, J., Ara, Y., Bogoch, I.I., Daneman, N., Matukas, L., McGeer, A., Powis, J., Schwartz, K.L., et al.: A platform for monitoring regional antimicrobial resistance using population-level antibiotic resistance data. The Lancet Infectious Diseases **16**(3), 334–340 (2016)

[20] Organization, W.H.: Who bacterial priority pathogens list 2024: bacterial pathogens of public health importance to guide research, development and strategies to prevent and control antimicrobial resistance (2024)

[21] Oviaño, M., Bou, G.: Rapid detection of antibiotic resistance in positive blood cultures by maldi-tof ms and an automated and optimized mbt-astra protocol for escherichia coli. Scientific Reports **9**(1), 1–8 (2020)

[22] Topol, E.: High-performance medicine: the convergence of human and artificial intelligence. Nature Medicine **25**(1), 44–56 (2019)

[23] Ventola, C.L.: The antibiotic resistance crisis: part 1: causes and threats. Pharmacy and Therapeutics **40**(4), 277–283 (2015)

[24] Weis, C.V., Jutzeler, C.R., Borgwardt, K.: Machine learning for microbial identification and antimicrobial susceptibility testing on maldi-tof mass spectra: a systematic review. Clinical Microbiology and Infection **26**(10), 1310–1317 (2020)

[25] Zankari, E., Hasman, H., Cosentino, S., Vestergaard, M., Rasmussen, S., Lund, O., Aarestrup, F.M., Larsen, M.V.: Identification of acquired antimicrobial resistance genes. Journal of Antimicrobial Chemotherapy **67**(11), 2640–2644 (2012)

Orthogonal Moments-Based Feature Extraction for MRI Classification

Gonzalo Degiuseppe$^{1[0009-0002-9397-9459]}$ and
Antonio Quintero-Rincón$^{1,2[0000-0003-0186-4049]}$

[1] Data Science Department, Data Science and AI Laboratory, Catholic University of Argentina (UCA), Argentina
[2] Computer Science Department, Catholic University of Argentina (UCA), Argentina
{gonzalodegiuseppe,antonioquintero}@uca.edu.ar

Abstract. Orthogonal moments are a current area of research in image analysis and pattern recognition. They are numerical values obtained by projecting an image intensity function onto a polynomial basis of the 2D coordinates to describe the distribution of pixels in an image space. This work proposes using orthogonal moments in MRI images as a feature extraction tool for detecting and classifying brain tumors, including gliomas, meningiomas, and pituitary cases. The method has three stages and employs the Random Forest model (RF) as its core foundation. In the first stage, Legendre Moments and the First and Second Order Chebyshev Moments are analyzed to extract features based on the weighted average of MRI image pixel intensities. In the second stage, the feature selection vector is calculated using the orthogonal moment features obtained in the previous stage. RF determines the majority vote for each class, while the Gini coefficient evaluates its concentration, leading to dimensionality reduction. In the final stage, the feature vector is utilized in a multiclass classifier framework based on RF to diagnose the type of brain tumor. The proposed methodology achieved an average accuracy of 96.49% across all brain tumor detection. Preliminary results indicate that this family of descriptors has significant potential for feature extraction in detecting brain tumors in MRI images.

Keywords: Brain tumor · Legendre Moment · Chebyshev Moment · Image analysis · Random Forest

1 Introduction

Brain tumors arise from the abnormal or uncontrolled proliferation of cells, leading to excess tissue masses. Tumors are generally divided into two primary categories: benign (noncancerous) and malignant (cancerous). While benign tumors do not contain cancerous cells, they may differ in growth rate and, in some cases, can reach significant sizes. These tumors do not metastasize to other parts of the body and can often be surgically removed with a good prognosis. In contrast, malignant tumors are characterized by cancerous cells that possess a high ability to invade nearby tissues. Their invasive characteristic complicates the definition of clear tumor margins, making treatment and surgical resection more challenging [1, 2]. This study analyzes three types of brain tumors: gliomas, the most common primary tumors of the brain and spinal cord, encompassing various subtypes with differing levels of aggressiveness and diagnostic complexity [3]; meningiomas, generally benign but potentially serious due to their intracranial location, often linked to factors such as hormonal influences or exposure to ionizing radiation [4]; and pituitary tumors, which develop in the pituitary gland, disrupting hormone production and potentially causing severe endocrine disorders [5]. As estimated by GLOBOCAN 2022, a report issued by the Global Cancer Observatory under the auspices of the World Health Organization, there were approximately 321,476 new cases of brain and central nervous system (CNS) tumors reported globally, in addition to 248,305 associated deaths. These tumors accounted for 2.6% of all newly diagnosed cancer cases and 2.5% of all cancer-related deaths. Regarding incidence, they ranked 19th among all cancer types, while in mortality, they were the 12th leading cause of cancer-related deaths [6]. The diagnosis of brain tumors is primarily based on neuroimaging techniques such as magnetic resonance imaging (MRI) and computed tomography (CT). However, definitive confirmation requires histopathological analysis of a biopsy sample [7, 8]. Given the significant impact of this disease on patient health, developing an effective early detection system is crucial. Accurate interpretation of MRI and CT images enables the identification of tumors at an early stage, facilitating timely and appropriate intervention, and ultimately improving patient outcomes.

Magnetic resonance imaging (MRI) has made significant progress with new techniques for medical image analysis. Among them, orthogonal moments which have long been used as quantitative measures of patterns present in images stand out for their ability to segment, classify, and reconstruct images. In recent years, orthogonal moments have been utilized in various medical imaging applications due to their capacity to efficiently encode image features while remaining robust against noise and geometric transformations. They have shown effectiveness in classification, segmentation, and image reconstruction, preserving essential structural information despite distortions or variations in image acquisition conditions [9]. Numerous investigations have explored orthogonal moments' capabilities within the medical image analysis domain. Thung et al. (2011) conducted a preliminary study comparing the compression efficiency and noise resilience of the Legendre and Chebyshev moments applied to X-ray images. The results indicated that Legendre moments provided superior noise resistance, while Chebyshev moments performed similarly to the Discrete Cosine Transform (DCT) in compression tasks. Their findings suggest that Legendre moments preserved image quality better under random white noise conditions, making them ideal for medical imaging applications where diagnostic accuracy is crucial [10]. Nallasivan G. and Subbiah (2017) analyzed computed tomography (CT) lung images using orthogonal moment-based texture features and applied segmentation methods, such as histogram analysis and watershed transformation, to extract diagnostic information [11]. Recent advancements have integrated deep learning techniques with orthogonal moments to enhance image analysis performance. Gao et al. (2024) proposed a hybrid approach combining fractional-order Chebyshev moments with deep neural networks for 3D image recognition. Their method demonstrated high classification accuracy across various datasets, leveraging moment invariants for robustness against scaling, rotation, and translation [12]. Di Ruberto et al. (2023) further validated the effectiveness of orthogonal moments for medical diagnosis by comparing their classification performance with convolutional neural networks (CNNs) across different datasets. Their analysis incorporated various classification models, including k-nearest neighbors (k-NN), Support Vector Machines (SVM), and Decision Trees (DT), among others. The study concluded that, despite CNNs' superior feature extraction capabilities, orthogonal moments provide competitive results with lower computational requirements, making them suitable for resource-constrained environments [13]. In Khalil et al. (2020), Legendre moments were utilized for 2D medical image classification, showing that they effectively preserved structural details and enabled high-precision classification [14]. Similarly, El Ogri et al. (2019) extended the application of discrete orthogonal moments to 2D and 3D medical images, supporting their utility in feature extraction and classification across different imaging modalities [15]. Beyond classification, orthogonal moments have also been used for medical image reconstruction. Hosny et al. (2013) examined the reconstruction of noisy medical images, confirming that orthogonal moments significantly enhanced image quality while preserving critical diagnostic features. Their experiments demonstrated that the proposed moment-based reconstruction method reduced reconstruction errors by 28% in comparison with traditional denoising techniques [16]. Additionally, orthogonal moments have been applied in image indexing and retrieval. Ahmadian et al. (2003) combined Gabor wavelets with Legendre moments to develop a hybrid indexing method, which resulted in a 25% improvement in retrieval efficiency compared to traditional indexing techniques [17].

This work aims to extract orthogonal moment features from MRI images for brain tumor classification. The public Kaggle database *Brain Tumor MRI Dataset* [18] has been used to accomplish this task. This dataset contains four categories of MRI images: glioma, meningioma, pituitary, and no-tumor cases. It's important to mention that this problem represents a multiclass classification framework. The method consists of three stages in the cascade and uses the Random Forest model (RF) as its core foundation. The first stage concentrates on orthogonal momentum-based feature extraction. Three orthogonal moments were individually evaluated, including the Legendre Moment, and the First and Second Order Chebyshev Moments. The feature extraction process (Φ) involves each orthogonal moment producing a matrix of values for each class based on the weighted average of the pixel intensities in the image. The second stage focuses on feature selection through dimensionality reduction from Φ, denoted as θ. RF computes the majority vote for each class, while the Gini coefficient identifies where the class is most concentrated, similar to statistical dispersion.

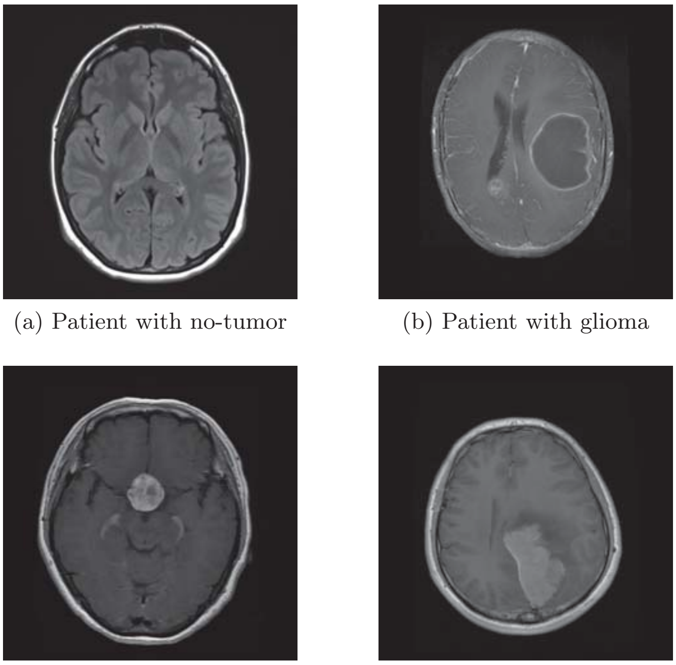
In the final stage, the feature vector θ is utilized in a multiclass RF classifier framework for diagnosing MRI images.

The rest of this document is organized as follows. Section 2 presents the proposed method in the following order: description of the data (Section 2.1), preprocessing (Section 2.2), Orthogonal moments and Moment features extraction (Sections 2.3, 2.4, and 2.5), Random Forest model (Section 2.6), Gini coefficient (Section 2.7), and feature vector (Section 2.8). In Section 3, results are analyzed and discussed. Finally, conclusions and perspectives are presented in Section 4.

2 Methology

2.1 Dataset

The Kaggle public database *Brain Tumor MRI Dataset* [18] was considered for experimentation. This dataset consists of 7023 human brain MRI images with the following cases: 1321 of glioma, 1339 of meningioma, 1457 of pituitary, and 1595 no-tumor cases, see Fig 1. The MRI images are arranged in separate folders for training and testing, which aids in model evaluation and performance assessment. The training folder comprises 80% of the dataset, while the testing folder contains the remaining 20%.



(a) Patient with no-tumor

(b) Patient with glioma

(c) Patient with pituitary

(d) Patient with meningioma

Fig. 1: Examples of MRI images: (a) No-tumor. (b) Glioma. (c) Meningioma. (d) Pituitary.

This work aims to extract orthogonal moment features from MRI images, including glioma, meningioma, pituitary, and no-tumor cases, for diagnosis using a multiclass Random Forest classification framework. A three-stage cascade method is proposed to achieve this, as shown in Figure 2. The first stage centers on orthogonal momentum-based feature extraction. The second stage emphasizes dimensionality reduction through feature selection by combining RF and the Gini coefficient. In the final stage, a multiclass RF classifier framework for MRI images is utilized for diagnosis. Note that the last two stages rely on RF as their core foundation. Next, the methods used in this work are introduced.

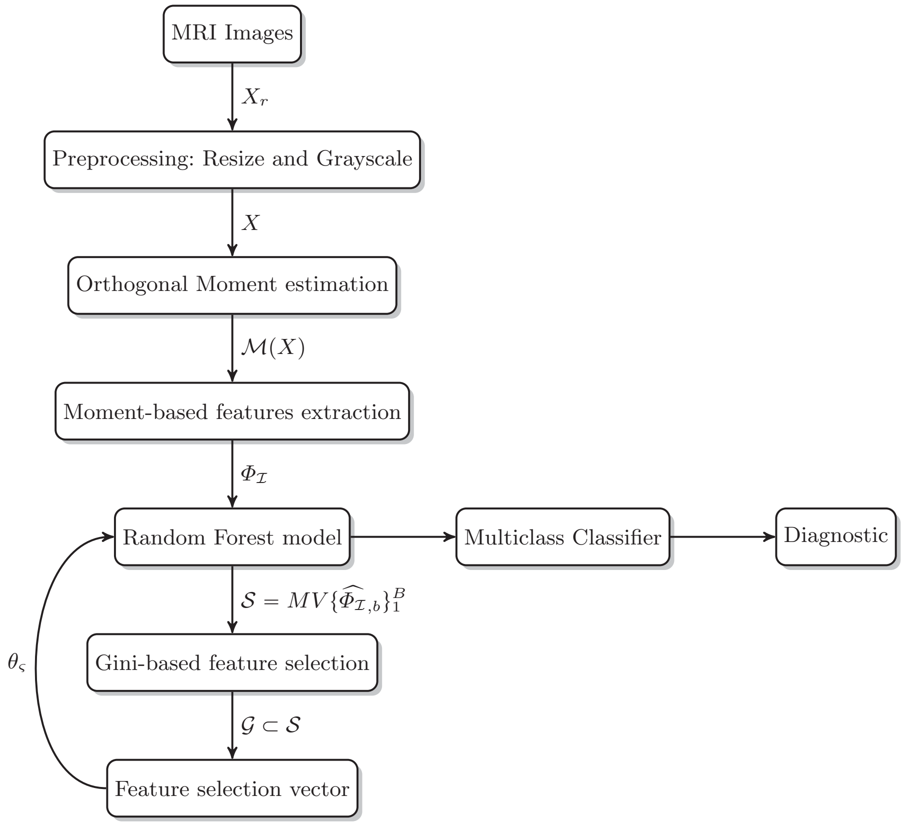

MRI Images
X_r
Preprocessing: Resize and Grayscale
X
Orthogonal Moment estimation
$\mathcal{M}(X)$
Moment-based features extraction
$\varPhi_{\mathcal{I}}$
Random Forest model
Multiclass Classifier
Diagnostic
$\mathcal{S} = MV\{\widehat{\varPhi_{\mathcal{I},b}}\}_1^B$
θ_ς
Gini-based feature selection
$\mathcal{G} \subset \mathcal{S}$
Feature selection vector

Fig. 2: Block diagram of the proposed method. MRI raw images (X_r) contain all classes ($\mathcal{I}$) under study, such as Class 1 = no-tumor cases, Class 2 = glioma cases, Class 3 = pituitary cases, and Class 4 = meningioma cases. X_r is resized and changed to grayscale yielding the X matrix. $\mathcal{M}(X)$ is the orthogonal moment-based feature extraction for each class, denoted as $\varPhi_{\mathcal{I}}$. RF computes the majority vote MV for each class $\mathcal{I}$ for each orthogonal moment, denoted as $\mathcal{S}$. Gini coefficient-based feature selection yields a subset from $\mathcal{S}$, $\mathcal{G} \subset \mathcal{S}$, denoted as θ. Finally, θ is classified to detect a brain tumor diagnosis using RF.

2.2 Preprocessing

Let $X_r \in R^{n \times m}$ be each raw MRI image matrix. Let M_x and M_y be the max size among all MRI images. The first step is resizing each MRI image to the dimensions (M_x, M_y) size using a cubic interpolation [19]. In the second step, all images are converted to grayscale. This process preserves the luminance while removing the hue and saturation information. These two preprocessing steps give a new image matrix X for each MRI image.

2.3 Moments

Let $f(x, y)$ be a two-dimensional Cartesian density distribution function that describes a grayscale X image content concerning its axes. Let $(p + q)$ be the order of a moment evaluated on the complete image plane ξ. The goal of a moment is to characterize the global and detailed geometric information about the image. Its general form is given by:

$$\mathcal{M}_{p,q} = \int\int_{\xi} \psi f(x, y) dx\ dy; \quad p, q = 0, 1, \cdots, \infty \tag{1}$$

where ψ is the weighting kernel or basis function that produces a weighted description of $f(x, y)$ over the entire image plane ξ.
Assume that ξ is divided into square pixels of dimensions 1×1, with constant intensity I over each square pixel. Let P_{xy} be the discrete pixel value defined as:

$$P_{xy} = I(x, y)\varDelta A \tag{2}$$

where $\varDelta A$ is the sample or pixel area equal to one. Then the discrete moment form is defined as:

$$\mathcal{M}_{p,q} = \sum_{x}\sum_{y} \psi P_{x,y}; \quad p, q = 0, 1, \cdots, \infty \tag{3}$$

2.4 Orthogonal moments

Orthogonal moments are defined by their minimal redundancy of information. This characteristic allows moments of different orders to describe unique information about an image, thus alleviating numerical problems linked to geometric and complex moments. Let y_p and y_q be two orthogonal functions over an interval $a \leqslant x \leqslant b$ then:

$$\sum_{a}^{b} y_p(x) y_q(x) dx = 0; \; p \neq q \tag{4}$$

The orthogonal moments studied in this work are introduced below.

Legrendre Moment (LM): The Legendre Moment is a complete orthogonal basis set defined over the interval $[-1,1]$. LM of order $(p+q)$ is defined as:

$$L_{pq} = \frac{(2p+1)(2q+1)}{4} \int_{-1}^{1} \int_{-1}^{1} P_p(x) P_q(y) f(x,y) dx\ dy \tag{5}$$

where P_p and P_q are the Legrende polynomials. For the orthogonality of the moments to be achieved, $f(x,y)$ must be defined over the identical interval as the basis set, where the p^{th} order Legendre polynomial is defined as

$$P_p(x) = \sum_k (-1)^k \frac{(2p-2k)!}{k!(p-k)!(p-2k)!} x^{p-2k} \tag{6}$$

For an image with current pixel P_{xy}, Eq. (5) become

$$L_{pq} = \frac{(2p+1)(2q+1)}{(N-1)^2} \sum_x \sum_y P_p(x) P_q(y) f_{xy} \tag{7}$$

LM defined regarding a recurrence relation with a uniform weight $\psi = 1$, is given by:

$$P_0(x) = 1, \quad P_1(x) = x \tag{8}$$

$$P_n(x) = \frac{(2n-1)xP_{n-1}(x) - (n-1)P_{n-2}(x)}{n}, \quad n \geqslant 2 \tag{9}$$

LM for a grayscale image is defined as:

$$\mathcal{M}_{LM} = \sum_x^m \sum_y^n I\ P_p(x) P_q(y) \frac{2}{m} \frac{2}{n} \tag{10}$$

To guarantee accurate implementation, the normalization of image coordinates is executed via the following transformation:

$$x_i = \frac{2i}{m-1} - 1, \quad y_j = \frac{2j}{n-1} - 1 \tag{11}$$

Discrete Chebyshev Moments (CHM) The Discrete Chebyshev Moments are a complete orthogonal basis set defined over the interval $[-1,1]$. CHM defined regarding a recurrence relation with a uniform weight $\psi = \frac{1}{\sqrt{1-x^2}}$, is given by:

$$T_0(x) = 1, \quad T_1(x) = x \tag{12}$$

$$T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x), \quad n \geqslant 2 \tag{13}$$

CHM is defined as:

$$\mathcal{M}_{CHM} = \sum_x^m \sum_y^n I\ T_p(x) T_q(y) \frac{2}{m} \frac{2}{n} \tag{14}$$

where $T_p(x)$ and $T_q(y)$ are the Chebyshev polynomials of the first kind evaluated at the image points.

Discrete Chebyshev Moments of the Second Order (CH2M) The Discrete Chebyshev Moments of the second order are a complete orthogonal basis set defined over the interval $[-1, 1]$. CH2M defined regarding a recurrence relation with a uniform weight $\psi = \sqrt{1 - x^2}$, is given by:

$$U_0(x) = 1, \quad U_1(x) = 2xU_n(x) = 2xU_{n-1}(x) - U_{n-2}(x), \quad n \geqslant 2 \tag{15}$$

CH2M is defined as:

$$\mathcal{M}_{CH2M} = \sum_{x}^{m} \sum_{y}^{n} I \; U_p(x) U_q(y) \frac{2}{m} \frac{2}{n} \tag{16}$$

where $U_p(x)$ and $U_q(y)$ are the Chebyshev polynomials of the second order evaluated at the image points.

we refer the reader to [9, 20, 21] for a comprehensive treatment of the mathematical properties of moments applied to image analysis.

2.5 Moment-based features extraction

Let X be the resized and grayscale MRI image. Let $N = \{N_1, N_2, \cdots, N_n\}$ be X with no-tumor cases, corresponding to class 1. Let $G = \{G_1, G_2, \cdots, G_n\}$ be X with glioma cases, corresponding to class 2. Let $P = \{P_1, P_2, \cdots, P_n\}$ be X with pituitary cases, corresponding to class 3. Let $M = \{M_1, M_2, \cdots, M_n\}$ be X with meningioma cases, corresponding to class 4. Let $\mathcal{I} \in N \cup G \cup P \cup M$ be any image from these four sets, with $\mathcal{I} \in \mathbb{R}^{r \times c}$, where r is the number of rows and c is the number of columns. Denoting $\mathcal{M}$ the orthogonal moment of $\mathcal{I}$ by $\widehat{\mathcal{M}}(\mathcal{I})$, the feature vector of $\mathcal{I}$ is defined as:

$$\Phi_{\mathcal{I}} = \left[\widehat{\mathcal{M}}(\mathcal{I})\right] \tag{17}$$

2.6 Random forest (RF) model

RF is a bagging classification and regression model that re-runs the same learning algorithm on different subsets of data to produce sufficiently diverse base models. RF combines multiple randomized decision trees and aggregates their predictions by averaging them [22]. Let $\widehat{\Phi}_{\mathcal{I},b} = \left[\widehat{\mathcal{M}}(\mathcal{I})\right]$ be the class prediction of the b-th random-forest tree. RF for classification is defined as:

$$\widehat{\Phi}^B(\mathcal{I}) = \mathcal{S} = \mathrm{MV} \left\{\widehat{\Phi}_{\mathcal{I},b}(\mathcal{I})\right\}_{b=1}^{B} \tag{18}$$

where B is the total number of trees used, $\widehat{\Phi}_b(\mathcal{I})$ is the predicted class from the b-th tree, and $\left\{\widehat{\Phi}_b(\mathcal{I})\right\}_{b=1}^{B}$ denotes the set of predictions from all trees in the RF. The final classification is determined by selecting the majority-voted class among all tree predictions. For more Random Forest details, see [23].

2.7 Gini-based feature selection

The Gini coefficient ($\mathcal{G}$) measures statistical dispersion and inequality to assess how well a feature separates data points into distinct classes. Feature importance scores are computed for each feature on a scale from 0 (perfect equality) to 1 (perfect inequality) as follows:

$$\mathcal{G} = 1 - \sum_{c} p^2_{\mathcal{S},c} \tag{19}$$

where p is the probability a random entry belongs to class c from the RF model majority vote $\mathcal{S}$, and $1 - p_c$ is the probability it would be misclassified.

2.8 Feature selection vector

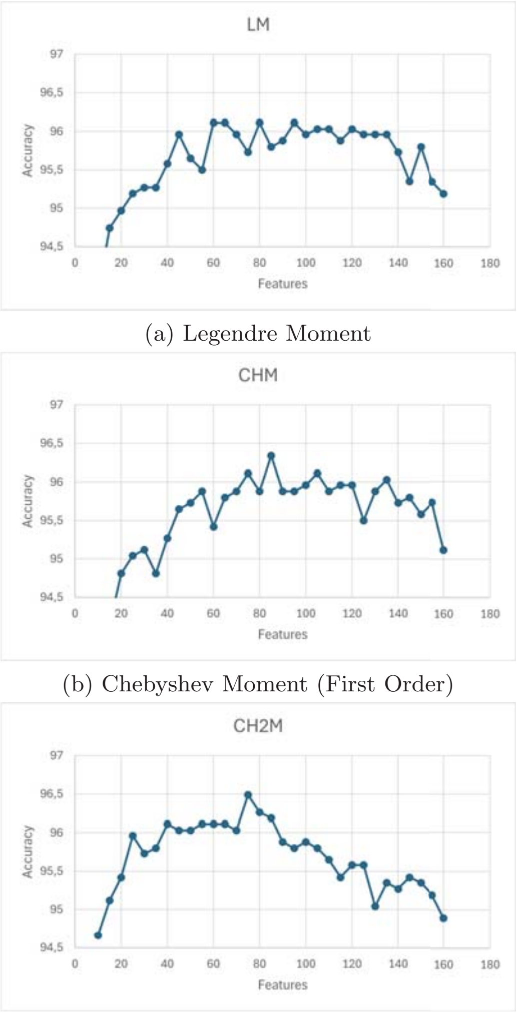
Let $\mathtt{N}, \mathtt{G}, \mathtt{P}, \mathtt{M}$ be the subsets resulting from applying the Gini coefficient $\mathcal{G}$ for all classes of MRI images obtained from Eq. (19). Let $\varsigma \in \mathtt{N} \cup \mathtt{G} \cup \mathtt{P} \cup \mathtt{M}$ be any image from these four subsets, with $\varsigma \in \mathbb{R}^{k \times l}$, where k is the number of rows and l is the number of columns. The feature vector subset of ς by $\widehat{\mathcal{G}}(\varsigma)$is:

$$\theta_\varsigma = [\widehat{\mathcal{G}}(\varsigma)] \tag{20}$$

3 Results and discussion

In this section, the evaluation results of the proposed method using the previously introduced database are reported. To recap, the dataset is divided into the following classes: 1321 glioma cases, 1339 meningioma cases, 1457 pituitary cases, and 1595 no-tumor cases. All MRI images ranged from 150 × 198 pixels to 1920 × 1080 pixels size. The maximum image size for each class was as follows: glioma (512×512 pixels), meningioma (1275×1427 pixels), pituitary (1365×1365 pixels), and no-tumor (1920×1080 pixels) During the preprocessing stage, all MRI images were resized to the maximum dimensions of each class using cubic interpolation [24]. Additionally, to maintain consistency in the MRI images, all images were converted to grayscale.

Next stage, three orthogonal moments, such as the Legendre Moment and the First and Second Order Chebyshev Moments, were studied to extract relevant features from the MRI images. The orders p and q were set in 12 empirically due to the high computational cost. At this point, each orthogonal moment produces a matrix $\mathcal{M} \in \mathbb{R}^{5712 \times 50}$ of values Φ for each class $\mathcal{I}$ as follows: $\Phi_{\mathcal{I}} = \mathcal{M}_{LM}(X)$, $\Phi_{\mathcal{I}} = \mathcal{M}_{CHM}(X)$, and $\Phi_{\mathcal{I}} = \mathcal{M}_{CH2M}(X)$. To dimensional reduction of each $\Phi_{\mathcal{I}}$, RF coupled with the Gini coefficient was used as a feature selection method. Fig 3 shows the evolution of each orthogonal moment. By visual inspection, it can be observed that the performance improves progressively as the number of features increases. However, after a certain value, increasing the features does not contribute significantly to the improvement of the model. Based on this analysis the following number of features were used in the feature selection vector θ: $\mathcal{M}_{LM} = 60$, $\mathcal{M}_{CHM} = 85$, and $\mathcal{M}_{CH2M} = 75$. Thus, every moment was reduced in $\mathcal{M}_{LM} = 64,71\%$, $\mathcal{M}_{CHM} = 50\%$, and $\mathcal{M}_{CH2M} = 55,88\%$. Remember that each feature selection θ is a subset of the orthogonal moment-based feature extraction. θ will be used in the RF multiclass classification framework.



(a) Legendre Moment

(b) Chebyshev Moment (First Order)

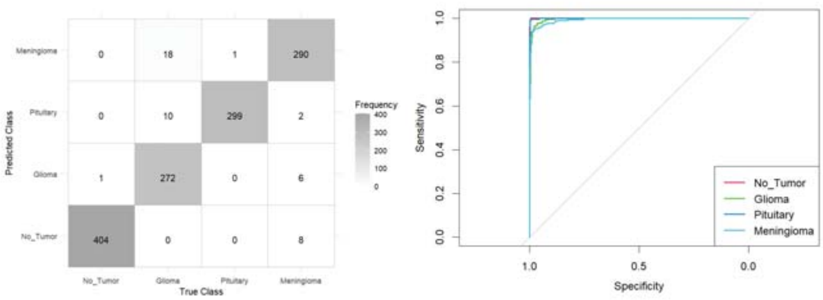
(c) Chebyshev Moment (Second Order)

Fig. 3: Gini feature selection performance for the different orthogonal moments.

Table 1 shows the performance metrics for the four classes and moments, including F1-Score, True Positive Rate (or recall, or sensitivity), Balanced Accuracy, and Area Under the Curve (AUC). Remarkably, the method proposed yields excellent results. The four classes: No-tumor, Glioma, Pituitary, and Meningioma have high detection scores, greater than 90% in all metrics throughout all orthogonal moments under study using the RF multiclass classification framework. For illustration, Fig 4 shows the Confusion matrix and receiver operating characteristic (ROC) curve for the best orthogonal moment, the Chebyshev moment (second order).

Class	Moment	F1-Score	TPR	B. Accuracy	AUC
No-Tumor	LM	99.01	99.51	98.53	0.99
	CHM	99.38	99.51	99.26	0.99
	CH2M	98.90	99.75	98.06	0.99
Glioma	LM	92.63	88.00	97.77	0.99
	CHM	91.80	87.66	96.34	0.99
	CH2M	93.96	90.66	97.49	0.99
Pituitary	LM	97.71	99.66	95.83	0.99
	CHM	97.55	99.66	95.53	0.99
	CH2M	97.87	99.66	96.14	0.99
Meningioma	LM	93.93	96.07	91.88	0.99
	CHM	93.76	95.75	91.85	0.99
	CH2M	94.31	94.77	93.85	0.99

Table 1: Performance metrics average for the four classes and moments under the multiclass RF model. F1-Score, True Positive Rate (or recall, or sensitivity), Balanced Accuracy, Area Under the Curve (AUC), Legendre Moment (LM), First (CHM) and Second Order (CH2M) Chebyshev Moment.



(a) Confusion matrix

(b) ROC curves

Fig. 4: Confusion matrix and receiver operating characteristic (ROC) curve for the best-performing model, the Chebyshev Moment (second order).

4 Conclusions

This work presents a new method based on orthogonal moments to classify and detect brain tumors in MRI images. The proposed method uses the Random Forest model (RF) as its core foundation. First, as data dimensionality reduction, combining it with the Gini coefficient technique as a feature selector. Second, it utilizes a multiclass classifier for brain tumor diagnosis. Legendre Moments and the First and Second Order Chebyshev Moments were analyzed to extract features from the MRI images. Performance metrics such as F1-Score, True Positive Rate, Balanced Accuracy, and Area Under the Curve were evaluated. Excellent high performance achieved an average accuracy of 96.49% in brain tumor detection, including glioma, meningioma, and pituitary cases.

In addition to its excellent performance, the proposed method based on orthogonal moment-based feature extraction in MRI images positions them as a family of descriptors with significant potential for feature extraction in detecting brain tumors in MRI images. The main limitation of the proposed method is that the orthogonal moments depend on their orders, which can generate a high computational cost.

Future work will focus on a more extensive evaluation of the proposed method, combining different orthogonal moments, and increasing tumor brain pathologies.

Bibliography

[1] National Institute of Neurological Disorders and Stroke (NINDS). Brain and spinal cord tumors. `https://www.ninds.nih.gov/health-information/disorders/brain-and-spinal-cord-tumors`, 2025. Accessed: 2025-02-02.

[2] Nimish A. Mohile and Alissa A. Thomas. *Brain Tumors. A Pocket Guide.* Springer, 2023.

[3] Ricky Chen, Matthew Smith-Cohn, Adam L. Cohen, and Howard Colman. Glioma subclassifications and their clinical significance. *Neurotherapeutics*, 14:284–297, 2017. https://doi.org/10.1007/s13311-017-0519-x.

[4] Joseph Wiemels, Margaret Wrensch, and Elizabeth B. Claus. Epidemiology and etiology of meningioma. *Journal of Neuro-Oncology*, 99:307–314, 2010. https://doi.org/10.1007/s11060-010-0386-3.

[5] Sylvia Asa and Shereen Ezzat. The pathogenesis of pituitary tumors. *Annual review of pathology*, 4:97–126, 02 2009. https://doi.org/10.1146/annurev.pathol.4.110807.092259.

[6] Freddie Bray, Mathieu Laversanne, Hyuna Sung, Jacques Ferlay, Rebecca L. Siegel, Isabelle Soerjomataram, and Ahmedin Jemal. Global cancer statistics 2022: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer Journal for Clinicians*, 74(3): 229–263, 2024. https://doi.org/10.3322/caac.21834.

[7] Anabel Blázquez López, Margarita Montes de Oca Carmenaty, Osmel Rodríguez Hernández, and Ricardo Leyva Tornés. Aspectos clínico-epidemiológicos de tumores del sistema nervioso central en pacientes pediátricos. Hospital Infantil Sur. Octubre 2015 - Octubre 2020. *EsTuSalud: Revista de Estudiantes de la Salud en Las Tunas*, 2(3), 2020.

[8] Xiaodong Li, Yawen Ma, Zirong Fan, and Rekha Khandia. *Brain Tumors: Advancements in Diagnostics and Innovative Therapies.* CRC Press, 2024.

[9] Jan Flusser, Tomas Suk, and Barbara Zitova. *2D and 3D Image Analysis by Moments.* Wiley, 2017.

[10] K.H. Thung, S.C. Ng, C.L. Lim, and P. Raveendran. A preliminary study of compression efficiency and noise robustness of orthogonal moments on medical X-Ray images. In *5th Kuala Lumpur International Conference on Biomedical Engineering 2011*, pages 587–590. Springer Berlin Heidelberg, 2011. https://doi.org/10.1007/978-3-642-21729-6_146.

[11] Nallasivan Gomathinayagam and Janakiraman Subbiah. Analysis of CT lung images using orthogonal moment features. *International Journal of Biomedical Engineering and Technology*, 24(2):121–132, 2017. https://doi.org/10.1504/IJBET.2017.084662.

[12] Lin Gao, Xuyang Zhang, Mingrui Zhao, and Jinyi Zhang. Recognition of 3D images by fusing fractional-order Chebyshev moments and deep neural networks. *Sensors*, 24(7):2352, 2024. https://doi.org/10.3390/s24072352.

[13] Cecilia Di Ruberto, Andrea Loddo, and Lorenzo Putzu. On the potential of image moments for medical diagnosis. *Journal of Imaging*, 9(3):70, 2023. https://doi.org/10.3390/jimaging9030070.

[14] Irshad Khalil, Sami Ur Rahman, Adnan Khalil, and Fakhre Alam. Two dimensional Legendre moments and its applications in classification of medical images. *Journal of Mechanics of Continua and Mathematical Sciences*, 15: 355–367, 2020. https://doi.org/10.26782/jmcms.2020.09.00028.

[15] Omar El Ogri, Achraf Daoui, Mohamed Yamni, Hicham Karmouni, Mhamed Sayyouri, and Qjidaa Hassan. 2D and 3D medical image analysis by discrete orthogonal moments. *Procedia Computer Science*, 148:428–437, 2019. https://doi.org/10.1016/j.procs.2019.01.055.

[16] Khalid M. Hosny, George A. Papakostas, and D. E. Koulouriotis. Accurate reconstruction of noisy medical images using orthogonal moments. In *2013 18th International Conference on Digital Signal Processing (DSP)*, pages 1–6, 2013. https://doi.org/10.1109/ICDSP.2013.6622675.

[17] Alireza Ahmadian, E. Faramarzi, and Sayadian. Image indexing and retrieval using Gabor wavelet and Legendre moments. volume 1, pages 560–563, 2003. https://doi.org/10.1109/IEMBS.2003.1279806.

[18] Brain tumor MRI dataset. `https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset`. Accessed: 2025-02-02.

[19] R.A. Becker, J.M. Chambers, and A.R. Wilks. *The New S Language: A Programming Environment for Data Analysis and Graphics*. Computer science series. Wadsworth & Brooks/Cole Advanced Books & Software, 1988. ISBN 9780534091927.

[20] Francisco Marcellàn and Walter Van-Assche. *Orthogonal Polynomials and Special Functions Computation and Applications*. Springer, 2006.

[21] S. M. Mahbubur Rahman, Tamanna Howlader, and Dimitrios Hatzinakos. *Orthogonal Image Moments for Human-Centric Visual Pattern Recognition*. Springer, 2019.

[22] Leo Breiman. Random Forests. *Machine Learning*, 45:5–32, 2001.

[23] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics. Springer, second edition, 2017.

[24] R. Keys. Cubic convolution interpolation for digital image processing. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 29(6):1153–1160, 1981. https://doi.org/10.1109/TASSP.1981.1163711.

Pipeline to detect spike-and-wave EEG patterns based on polynomial regression modeling and Taylor series feature selection

Matias F. Adell[1], Javier Balda[1], Facundo Casas[1], Carlos D'Giano[3], and Antonio Quintero-Rincón[1,2,3][0000−0003−0186−4049]

[1] Department of Data Science, Data Science and AI Laboratory, Catholic University of Argentina (UCA), Buenos Aires, Argentina.
[2] Department of Computer Sciences, Catholic University of Argentina (UCA), Buenos Aires, Argentina.
[3] Epilepsy and Telemetry Integral Center, Foundation for the Fight against Pediatric Neurological Disease (FLENI), Buenos Aires, Argentina.
{matiasadell,javierbalda,facundocasas,antonioquintero}@uca.edu.ar

Abstract. Epilepsy is a common neurological disorder diagnosed and monitored through EEG recordings. Accurate spike-and-wave (SW) pattern classification is crucial for distinguishing this epileptic seizure disorder from normal brain wave activity (NW). However, mathematically modeling SW remains challenging, affecting classification accuracy. This study proposes a pipeline in two stages combining polynomial regression techniques, and data processing, in a machine-learning classification scheme. At the first stage of decision-making, the idea is to create a generalized waveform mother that represents all the waveforms of the EEG patterns, such as SW and NW. This waveform is derived from a polynomial regression model that is assessed by the truncation error of the Taylor series. In the second stage, a feature selection algorithm based on a vector that includes the coefficients from Taylor and the statistical properties of the SW and NW waveforms was designed for the machine learning classifier. This algorithm uses the confidence interval to extract the Taylor series points that do not represent the generalized mother equation. This yields a dimensional reduction of this vector, which can be used in a classification and detection scheme. Three polynomial regression models, such as Fourier, Gaussian, and sums-of-sines were evaluated using the pipeline methodology. The best model was the Fourier regression, which achieved an accuracy of 96.2% using the SVM classifier with a Gaussian kernel to detect spike-and-wave patterns.

Keywords: Spike-and-wave · Polynomial regression · Taylor series · Feature selection

1 Introduction

Epilepsy is one of the most common neurological diseases, affecting approximately 50 million people worldwide [1]. This condition is characterized by the occurrence of epileptic seizures, which are the result of abnormal and excessive electrical activity in the brain. The electroencephalogram (EEG) is a crucial biomedical tool used for the diagnosis and treatment of epilepsy because it allows for the recording and analyzing of brain waves to detect epileptiform activity, such as the spike-and-wave (SW) waveform pattern [2]. In the healthcare industry, EEG signals are widely used for detecting and classifying epileptiform waveform patterns, essential for an accurate diagnosis [3]. However, signal interpretation remains challenging due to the complexity of the waveforms and the need to differentiate between epileptiform activity and normal brain waves (NW) [4]. Automating this process using machine learning techniques and advanced signal processing has gained ground in the last decade, improving the accuracy and efficiency of diagnosis [5, 6]. However, difficulties persist in the precise mathematical representation of SW patterns and in assessing the impact of errors in this representation on classification models [7]. The problem lies in the need to develop mathematical models in EEG analysis that accurately capture the shape of SW waves, allowing them to be differentiated from NW and improving the classification algorithms' accuracy [8]. Today, the lack of precision in waveform representation can lead to significant errors in classification, directly affecting the ability of automatic systems to make reliable diagnoses [9, 10, 11, 12, 13]. This study addresses this problem by implementing a comprehensive pipeline that combines regression techniques, data processing, and classification models to analyze SW and NW waveforms. This pipeline, the main contribution of this work, consists of two stages. The first stage is for decision-making, and the second is for feature selection and classification. The decision-making goal is to use polynomial regression modeling to create a generalized mother equation based on the EEG waveforms. This generalized waveform is asses through the truncation error of the Taylor series. This stage produces a polynomial regression function and the Taylor coefficients at each point of this function. In this study, Fourier regression was the best model regarding the other two models studied, such as Gaussian, and Sum-of-Sines. The second stage applies the optimal results from the first stage to detect and classify spike-and-wave epileptiform patterns in EEG signals based on the feature selection algorithm. The input of this algorithm is the feature vector given by mean, median, standard deviation, kurtosis, and skewness from the SW and NW waveforms, and the Taylor series coefficients in each point. This algorithm focuses on extracting the Taylor series points that are not representative from the generalized mother equation using the confidence interval, yielding a dimensional reduction of this vector to be used in a classification and detection scheme.

The rest of this document is organized as follows. Section 2 presents the database, the mathematical theory, the proposed pipeline, and the feature selection algorithm. In Section 3, results are analyzed and discussed. Finally, conclusions and perspectives are presented in Section 4.

2 Material and Methods

2.1 Database

Signals were acquired from 12 patients at the Epilepsy Department of the Foundation for the Fight Against Pediatric Neurological Diseases (FLENI). An expert neurologist in epilepsy labeled 339 SW and 441 NW waveforms of the EEG signals, indicating the onset and duration of the epileptic waveform. A standard 10-20 EEG system with a sampling rate of 256 Hz was used with the following 22 channels: Fp1, Fp2, F7, F3, Fz, F4, F8, T3, C3, Cz, C4, T4, T5, P3, Pz, P4, T6, O1, O2, Oz, FT10, and FT9. Each waveform consists of a temporal sequence of amplitudes with morphological characteristics. SW waveforms are characterized by their regular and symmetrical morphology, combining spike peaks and smoother waves. This distinguishes them from NW waveforms, which have a less structured and more variable morphology. On average, the SW amplitudes are approximately 500, while the NW amplitudes are around 300. The total duration of the waveforms is approximately 100 to 264 seconds. Figure 1 shows representative examples of both types of waveforms. See [14] for more details of this database.

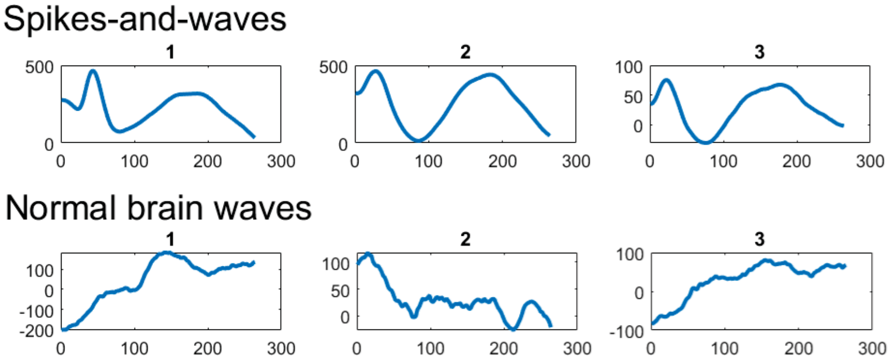


Fig. 1. Spike-and-waves and normal brain waves examples.

2.2 Pipeline methodology

The pipeline proposed in this study addresses the classification of spike-and-waves (SW) and normal brain waves (NW) using a comprehensive approach that combines data processing, polynomial regression techniques, and the Taylor series in a classifier scheme. The pipeline consists of two stages. The first stage is for decision-making, illustrated by the red dashed line in Figure 2. The second stage applies the optimal results from the first stage to feature selection and detect and classify spike-and-wave epileptiform patterns in EEG signals, illustrated by the blue dashed line in Figure 2.

The first stage begins with resizing the waveforms because all signals have different total durations in seconds. Thus, the maximum size of all waveforms was calculated, and each SW and NW signal was resized to this maximum size using linear interpolation (Section 2.3). This process results in a single resolution size for all signals. Since the equation of the waveforms of interest is not known in advance, three polynomial regression models (Section 2.4) such as Fourier, Gaussian, and Sum-of-Sines were evaluated according to the metrics to identify which model best fits each resized waveform. The metrics used, such as Degrees of freedom for Error, Coefficient of determination R^2, Adjusted R^2, and the Root mean square error (Section 2.9), yielded that the Fourier regression was the best model. It produces an equation with coefficients representing each waveform. These coefficients were averaged to calculate a single overall coefficient to estimate the generalized mother waveform equation (Section 2.5). Finally, the Taylor series (Section 2.6) was utilized to approximate the equation of the generalized mother waveform at each point. The underlying idea is to assess how the series behaves relative to the original waveform and to determine whether it will be a good representation for subsequent analysis in the second stage of the pipeline. In addition, the truncation error (Section 2.7) is calculated at each point and cumulatively, providing a measure of the accuracy of the Taylor approximation for representing these waves.

In the second stage of the pipeline, a feature vector with two sets was built. The first set is based on the Taylor Series evaluation of degree 8 at each point. This indicates that the series is calculated in 1-second intervals, fully encompassing each SW and NW signal. Note that this data represents the results of the best decision-making model, the Fourier regression. The second set uses classical statistical properties, extracted directly from the original signals. The mean, median, standard deviation, kurtosis, and skewness improve the information available for the analysis.

At this point, the feature vector contains the Taylor Series approximation points and the statistical properties of each SW and NW signal (Section 2.10). This feature vector carries all the information needed to detect and classify spike-and-wave epileptiform patterns in EEG signals. The feature selection of this feature vector was performed using the proposed algorithm 1. This algorithm analyzes the points of the generalized mother equation that do not represent the classification model using the confidence interval. Subsequently, all features are normalized using the Min-Max Scaling technique from -1 to 1, ensuring that the data are in a uniform range and comparable (Section 2.8). To validate the effectiveness of the dataset, a 5-fold cross-validation is implemented, reserving 20% of the data for final testing. The theoretical framework used in the pipeline is introduced below.

2.3 Linear interpolation

Let $S_{max} = \max(\max(\mathrm{SW}), \max(\mathrm{NW}))$ be the maximum size of all waveforms. Let t_i and t_{i+1} be two successive points from each vector related to each waveform or class, $\mathrm{SW}(t)$ and $\mathrm{NW}(t)$. The goal is to find an intermediate point t between these two points. Then the linear 1D interpolation correspondent to each interval, $t_{i+1} - t_i$, and for $1 \leq \mathrm{SW}(t)|\mathrm{NW}(t) \leq S_{max}$ is given by:

$$t = (1-t) * t_i + t * t_{i+1} = t_{i+1} + t(t_{i+1} - t_i) \tag{1}$$

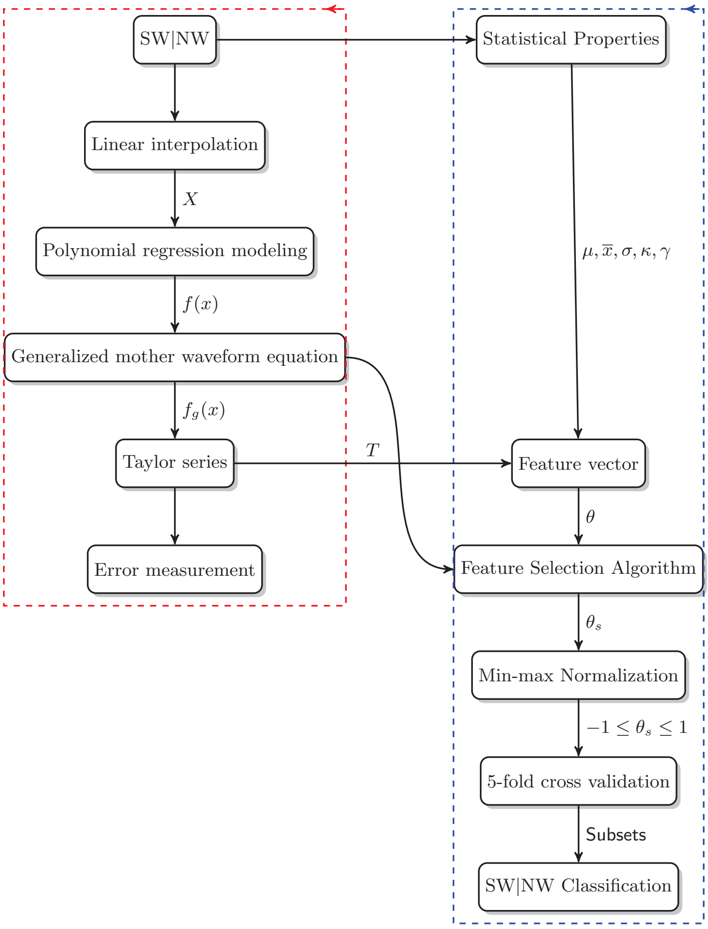


Fig. 2. Pipeline to detect and classify spike-and-wave epileptiform patterns in EEG signals. The decision-making stage includes all processes within the red dashed line. The feature selection and classification stage includes all processes within the red dashed blue.

2.4 Polynomial regression modeling:

Since the mathematical waveform of spike-and-wave is not known in advance, a mathematical model was created to describe its characteristics. Gaussian, Fourier, and Sum-of-sines polynomial regression models were evaluated to fit the EEG signals. These models can capture the specific and regular morphology of the waveforms, which is important for accurate pattern recognition. Each one is introduced below.

Gaussian Regression: A statistical method that uses the Gaussian function to describe a relationship between waveforms, approximating the function to fit the peaks. The Gaussian model is expressed as:

$$f(x) = \sum_{i=1}^{n} a_i \exp\left[-\left(\frac{x-b_i}{c_i}\right)^2\right] \tag{2}$$

where a_i is the peak height or amplitude, b_i is the peak's center position or location, c_i is the peak width, and n is the number of peaks to fit.

Fourier Regression: A statistical method that uses the Fourier series to describe a relationship between waveforms as a sum of sine and cosine functions. The trigonometric Fourier series is given by:

$$f(x) = a_0 + \sum_{i=1}^{N} a_i \cos(i\omega x) + b_i \sin(i\omega x) \tag{3}$$

where a_0 is the intercept, a constant term associated with the $i = 0$ cosine term, a_i and b_i are the Fourier coefficients, n is the number of terms, and ω is the fundamental frequency.

Sum-of-sines regression: A statistical method that fits a weighted sum-of-sines functions to data. The mathematical expression for this model is:

$$f(x) = \sum_{i=1}^{n} a_i \sin(b_i x + c_i) \tag{4}$$

where a_i, b_i, and c_i are adjustable parameters that control the amplitude, frequency, and phase of each sin component respectively, and n is the number of terms. Note that this model includes the phase constant, and does not include the intercept term. This is the main difference from the Fourier Regression method.

2.5 Generalized mother waveform equation

Let $f(x)$ be the polynomial regression model that best fits each X EEG waveform. Let $\mathcal{C} \in R^{f \times c}$ the coefficients' matrix from each $f(x)$, where f is each equation and c each coefficient. The mean of all coefficients was calculated to yield a generalized representative alignment, which captures the studied waveforms' main periodic and morphological characteristics.

2.6 Taylor approximation

The Taylor series expansion provides a way to represent a function as an infinite sum of terms. Its derivatives are calculated at a specific point to approximate complex functions to feasible polynomials [15]. The idea is to approximate the generalized alignment waveform of the set of SW and NW to a polynomial function. In this case, the best polynomial regression model that fits the EEG patterns, see Section 2.4. Thus, it is necessary to define the function to be expanded, the variable, the initial point, and the number of terms in the Taylor series. :

$$f(x_{i+1}) = \sum_{n=0}^{\infty} \frac{f^{(n)}(x_i)}{n!}(x_{i+1} - x_i)^n \tag{5}$$

where $f^{(n)}(x_i)$ is the n-derivative of f evaluated at x_i, and x_i is the point around which the function is expanded.

2.7 Error Measurement

It is used to assess the accuracy of an approximation or model compared to the true value. It helps determine how close or far an estimated result is from the true value, allowing for improved methods and informed decisions. Different metrics are used for this purpose.

True Error ($\boldsymbol{E_t}$) : It is the difference between the exact value of a function or series and its finite approximation. It is the error generated by putting a finite number of decimals in an approximation.

$$E_t = \text{true value} - \text{approximate value} \tag{6}$$

Percent Relative Error ($\boldsymbol{\epsilon_t}$) : It indicates how significant the difference is between the prediction and the actual value.

$$\epsilon_t = \frac{E_t}{\text{true value}} 100\% \tag{7}$$

Normalized Percent Error ($\boldsymbol{\epsilon_a}$) : It measures errors when the actual approximation value is unknown.

$$\epsilon_a = \frac{\text{present approximation} - \text{previous approximation}}{\text{present approximation}} 100\% \tag{8}$$

Truncation error $\boldsymbol{E_\xi}$: It arises from using the Taylor series approximation instead of an exact mathematical expression method. The complete expansion of the Taylor series Eq. (5) is defined as

$$f(x_{i+1}) = f(x_i) + f'(x_i)h + \frac{f''(x_i)}{2!}h^2 + \cdots + \frac{f^{(n)}(x_i)}{n!}h^n + R_n \tag{9}$$

$$R_n = \frac{f^{(n+1)}(\xi)}{(n+1)!}h^{n+1} \tag{10}$$

where the subindex n of R indicates the residue of the n order approximation, ξ is the truncation error, a value of x that is located somewhere between x_i and x_{i+1}. For a comprehensive mathematical treatment of truncation errors, see [15].

2.8 Min-max normalization

It is a method for scaling data to a fixed range of values from minimum to maximum. It is beneficial to prevent data analysis from being influenced by the variation in time.

$$x' = \frac{2 \times (x - \min(x))}{\max(x) - \min(x)} - 1 \tag{11}$$

where x is the original value of the wave, $\min(x)$ is the minimum value of the wave in the data set, $\max(x)$ is the maximum value of the wave in the data set, and x' is the normalized wave value, scaled in the range -1 to 1.

2.9 Metrics performance

The following metrics were used to evaluate the fit quality of the regression models used in this study:

Sum of Squares Error (SSE): It measures the discrepancy between the observed values and the values predicted by the regression model. It is calculated by summing the squares of the differences between the actual values y_i and the predicted values $\hat{y}_i$ [16]. A lower SSE indicates a better fit of the model to the data. It is given by:

$$SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{12}$$

Degrees of Freedom for Error (DFEs): It represents the number of independent observations in a model minus the number of estimated parameters, including the intercept. DFEs are primarily used in assessing the statistical significance of regression coefficients because they influence the error variance estimation. It is given by:

$$DFE = n - p - 1 \tag{13}$$

where n is the total number of observations and p is the number of predictors in the model [17].

Coefficient of Determination $\mathbf{R^2}$: It measures the proportion of the variance in the dependent variable explained by the regression model. R^2 ranges between 0 and 1, where a value of 1 indicates that the model perfectly explains the variability observed in the data. It is given by

$$R^2 = 1 - \frac{SSE}{SST} \tag{14}$$

where SST is the Total Sum of Squares, representing the total variability in the data, although R^2 provides a general measure of model fit [18].

Adjusted $\mathbf{R^2}$: It is a modified version of the R^2 that takes into account the number of predictors in the model. Unlike R^2, adjusted R^2 penalizes adding additional predictors that do not significantly improve the model. It is calculated as:

$$R^2_{adj} = 1 - \frac{(1 - R^2)(n - 1)}{n - p - 1} \tag{15}$$

where n is the number of observations, and p is the number of predictors [19].

Root Mean Square Error (RMSE): RMSE measures the average magnitude of the error in the model's predictions. It is the square root of the average of the squared errors and is expressed in the same units as the dependent variable, which makes it easier to interpret [20]:

$$RMSE = \sqrt{\frac{SSE}{n}} \tag{16}$$

2.10 Feature vector

Let $\mu, \overline{x}, \sigma, \kappa, \gamma$ be the mean, median, standard deviation, kurtosis, and skewness, the statistical properties respectively of each SW and NW waveform. Let $T' \in R^{p \times s}$ be the vector of p points and s seconds corresponding to the Taylor Series approximation points in each sec. The final feature vector is defined as

$$\theta = [T', \mu, \overline{x}, \sigma, \kappa, \gamma] \tag{17}$$

2.11 Feature Selection

In this paper, a feature selection algorithm was designed to reduce the number of variables in the dataset, see Algorithm 1. Remember that the dataset contains the Taylor approximation points and the statistical properties. For each second, a Taylor approximation point was computed. Thus, each point represents a feature. This algorithm focuses on reducing these points because not all points are equally representative of the overall signal. Therefore, those points that do not provide significant information are identified and eliminated. For this purpose, the algorithm compares the waveforms between the generalized equation and the polynomial regression model of each SW and NW pattern, identifying the points inside and outside the confidence interval. The points within the confidence interval are considered redundant because they do not provide new or significant information about the signal. These points are marked for elimination. Therefore, the points of interest are those outside the confidence interval, see Figure 3. This process allows a dimensional reduction of the dataset without losing the most relevant aspects of the signal.

Algorithm 1 Feature selection algorithm

```
Input: $f(x)$ (regression), $z$ (Confidence interval), $n$ (Regression duration),
Output: $cols$ (Non-representative points of the regression)
$cols \leftarrow []$
$i \leftarrow 0$
$z \leftarrow 0.2$ // Choice according to the criterion
while $i < n$ do
    $int\_sup \leftarrow f(i) + z \cdot \sigma$ //$\sigma$ is the std
    $int\_inf \leftarrow f(i) - z \cdot \sigma$
    for $j \leftarrow i + 1 : n$ do
        if $f(j) \leq int\_sup$ and $f(j) \geq int\_inf$ then
            $cols \leftarrow j$
        else
            break
        end if
    end for
    $i \leftarrow j$
end while
return $cols$
```

3 Results

This section presents the results from the proposed pipeline for detecting and classifying spike-and-wave epileptiform patterns in EEG signals. Three polynomial regression models in the decision-making pipeline, namely Fourier, Gaussian, and sum-of-sines, were evaluated with their metrics, to select the best model that fits the EEG signal waveform. Table 1 shows the performance metrics for the three polynomial regression models evaluated. Fourier regression stands out for its superiority in all the assessed metrics. A higher R^2 and lower RMSE values suggest a better fitting capacity and accuracy in representing SW waveforms. Gaussian regression also shows acceptable performance, with relatively high value for R^2 and adjusted R^2. In contrast, the Sum-of-sines regression presented significantly inferior performance, with a negative R^2 value and a high RMSE value, suggesting a lack of ability to capture the characteristics of SW waveforms. The Fourier regression was fitted for each EEG waveform from the dataset. Remember that the dataset contains 339 SW and 441 NW signals, see Section 2.1. This process yields 18 coefficients for each waveform. These are averaged to generate a final generalized mother waveform equation, see equation (18) with a morphology similar to that observed in the original signals, see signal with blue color in Figure 3.

Table 1. Comparison of the three Polynomial regression models evaluated

Model	**SSE**	**DFE**	**R^2**	**Adjusted R^2**	**RMSE**
Gauss	755346.3235	155.8053	0.7367	0.7089	53.6163
Fourier	320986.8081	152.7788	0.9093	0.9020	26.5285
Sum-of-sines	825650.9445	146.7876	0.2540	-0.0936	64.9567

$$\begin{aligned} f_{\text{SWS}}(x) = {} & 38.2296 + 14.6907\cos(0.0312\cdot x) - 134.9640\sin(0.0312\cdot x) \\ & - 95.5140\cos(0.0624\cdot x) - 4.6392\sin(0.0624\cdot x) \\ & - 32.4604\cos(0.0936\cdot x) + 55.7023\sin(0.0936\cdot x) \\ & + 30.1549\cos(0.1248\cdot x) + 4.1776\sin(0.1248\cdot x) \\ & + 17.1263\cos(0.156\cdot x) - 22.0911\sin(0.156\cdot x) \\ & + 5.1722\cos(0.1872\cdot x) - 9.7515\sin(0.1872\cdot x) \\ & + 6.1877\cos(0.2184\cdot x) - 1.8771\sin(0.2184\cdot x) \\ & + 4.3840\cos(0.2496\cdot x) + 0.1376\sin(0.2496\cdot x) \end{aligned} \tag{18}$$

Note that different Taylor series degrees generate good coefficients for a classifier and detection scheme with low computational complexity and small errors. The Taylor series approximation errors were calculated at each point of the generalized Fourier function with the average of the coefficients. Remember that these values are part of the input from feature vector Eq. 17 for the second stage of the pipeline. Table 2 shows the cumulative errors of all signal points in each function degree. Note that, in both Figures, as the degree of the Taylor series approximation grows until n=8, the error decreases until it becomes imperceptible.

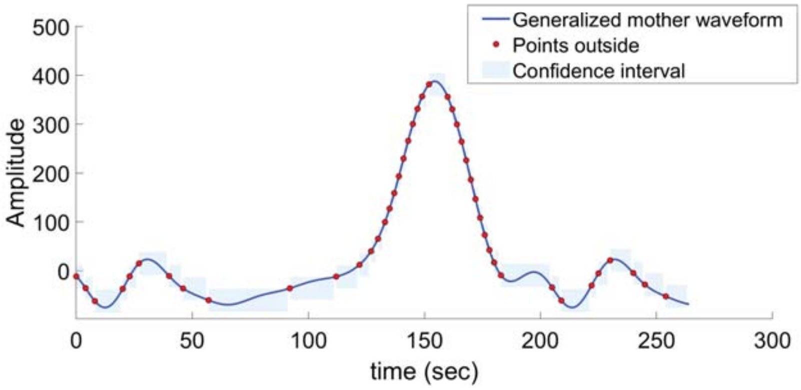


Fig. 3. Graphical representation of the generalized mother waveform (blue color) from the Fourier regression, with its confidence interval of $z = 0.2$ (light blue rectangles) and its points outside the confidence intervals from the Taylor series approximation (red circles).

Table 2. Cumulative errors. E_t: True error. E_ξ: Truncation error. E_t:Percent relative error. ϵ_a: Normalized percent error

Taylor degree	E_t	E_ξ	E_t	ϵ_a
0	-60.7772	-77.4907	-7.7491e+05	1.5266e+04
1	1.1755	-2.8812	1.0822e+03	9.0007e+03
2	0.2302	0.5151	66.0135	2.4176e+04
3	-6.2901e-04	-0.0066	0.1901	1.5160e+04
4	-5.2751e-04	-0.0011	0.1765	1.5225e+04
5	-1.6621e-06	2.3260e-05	0.0076	1.5235e+04
6	5.9759e-07	1.1875e-06	1.6747e-04	1.5235e+04
7	4.0414e-09	-2.8451e-08	2.9340e-06	1.5235e+04
8	-4.4729e-10	-7.9035e-10	1.2316e-07	1.5235e+04

For the second stage of the pipeline, the feature vector contains the Taylor series approximation points T of degree 8 at each point, with the statistical properties $\mu, \overline{x}, \sigma, \kappa, \gamma$ of the SW and NW signals. The proposed feature selection algorithm 1 achieved a dimensionality reduction of 23%. For illustration, Figure 3 shows the confidence interval (light blue rectangles), the points T outside the confidence interval (red color) from the generalized Fourier function (blue color). Remember that for the feature selection algorithm θ_s, the points of interest are those outside the confidence interval. θ_s is the feature selection from the feature vector θ, this vector was normalized to be tested in a machine-learning scheme.

Three classical machine models, such as Decision Trees, SVM with Gaussian kernel, and 10-nearest neighbors, were tested with θ_s. Table 3 shows the variations of the classification models depending on the degree of Taylor approximation in terms of accuracy. All models perform well, but the Gaussian SVM excels compared to the other models as the Taylor degree increases.

Table 3. Classification models comparison in terms of accuracy.

Taylor degree	Decision Tree	SVM	10-NN
0	0.8970	0.9310	0.8140
1	0.9420	0.9620	0.8330
2	0.8910	0.9290	0.8190
3	0.8900	0.9280	0.8200
4	0.8900	0.9300	0.8200
5	0.8890	0.9290	0.8210
6	0.8870	0.9280	0.8200
7	0.8860	0.9280	0.8210
8	0.8860	0.9300	0.8200

4 Conclusions

This work proposed an original two-stage pipeline to classify spike-and-wave epileptiform patterns in EEG signals. The first stage is for decision-making and the second is for feature selection and classification. At the decision-making stage, polynomial regression models of Fourier, Gaussian, and sums-of-sines were analyzed to find the best model that fits all the EEG waveform patterns, such as SW and NW. The best model was Fourier regression based on error metrics. From this model, a generalized waveform equation was computed, averaging all its coefficients for all waveform patterns. This generalized equation was evaluated through the truncation error of the Taylor series. In the second stage, a feature selection algorithm was designed. The algorithm computes the confidence interval for the generalized equation and the Taylor coefficients given by the polynomial regression model of each SW and NW pattern. The points inside and outside the confidence interval are detected and compared. Only the points outside the confidence interval were considered to yield a dimensional reduction of this data. Finally, the algorithm output coupled with the statistical properties of the SW and NW waveforms builds a vector to be used in a classification and detection scheme. The Fourier regression achieved an accuracy of 96.2% using the SVM classifier with a Gaussian kernel, allowing the detection of spike-and-wave patterns.

In addition to its excellent performance, the proposed pipeline has a low computational cost. The proposed pipeline's main limitation is that it does not explicitly consider physiological and non-physiological artifacts. Future work will focus on evaluating the proposed pipeline more extensively and studying robust feature extraction methods using highly imbalanced data.

Bibliography

[1] World health organization. epilepsy. https://www.who.int/news-room/fact-sheets/detail/epilepsy, 2025. Accessed: 2025-02-02.

[2] Bruce J. Fisch. *Epilepsy and Intensive Care Monitoring: Principles and Practice*. Demos Medical, 2009.

[3] J. Gotman. Automatic recognition of epileptic seizures in the EEG. *Electroencephalography and Clinical Neurophysiology*, 54(5):530–540, 1982. https://doi.org/10.1016/0013-4694(82)90038-4.

[4] Jonathan J. Halford. Computerized epileptiform transient detection in the scalp electroencephalogram: Obstacles to progress and the example of computerized ECG interpretation. *Clinical Neurophysiology*, 120(11):1909–1915, 2009. https://doi.org/10.1016/j.clinph.2009.08.007.

[5] U. Rajendra Acharya, S. Vinitha Sree, G. Swapna, Roshan Joy Martis, and Jasjit S. Suri. Automated EEG analysis of epilepsy: A review. *Knowledge-Based Systems*, 45:147–165, 2013. https://doi.org/10.1016/j.knosys.2013.02.014.

[6] Ali H. Abdulwahhab, Alaa Hussein Abdulaal, Assad H. Thary Al-Ghrairi, Ali Abdulwahhab Mohammed, and Morteza Valizadeh. Detection of epileptic seizure using EEG signals analysis based on deep learning techniques. *Chaos, Solitons & Fractals*, 181:114700, 2024. https://doi.org/10.1016/j.chaos.2024.114700.

[7] Risto J. Ilmoniemi and Jukka Sarvas. *Brain Signals: Physics and Mathematics of MEG and EEG*. The MIT Press, 2019.

[8] A. Ananthi, M.S.P. Subathra, S. Thomas George, Geno Peter, Albert Alexander Stonier, and N.J. Sairamya. A fusion wavelet-based binary pattern approach for enhanced electroencephalogram signal classification. *Computers and Electrical Engineering*, 123:110019, 2025. https://doi.org/10.1016/j.compeleceng.2024.110019.

[9] Ali Shahidi Zandi, Manouchehr Javidan, Guy A. Dumont, and Reza Tafreshi. Automated real-time epileptic seizure detection in scalp EEG recordings using an algorithm based on wavelet packet transform. *IEEE Transactions on Biomedical Engineering*, 57(7):1639–1651, 2010. https://doi.org/10.1109/TBME.2010.2046417.

[10] Yannick Roy, Hubert Banville, Isabela Albuquerque, Alexandre Gramfort, Tiago H Falk, and Jocelyn Faubert. Deep learning-based electroencephalography analysis: a systematic review. *Journal of Neural Engineering*, 16 (5):051001, 2019. https://doi.org/10.1088/1741-2552/ab260c.

[11] Aradia Fu and Fred A Lado. Seizure detection, prediction, and forecasting. *Journal of Clinical Neurophysiology*, 3(41):207–213, 2024. https://doi.org/10.1097/WNP.0000000000001045.

[12] Brandon M. Brown, Aidan M. H. Boyne, Adel M. Hassan, Anthony K. Allam, R. James Cotton, and Zulfi Haneef. Computer vision for automated

seizure detection and classification: A systematic review. *Epilepsia*, 65(5): 1176–1202, 2024. https://doi.org/10.1111/epi.17926.

[13] N. Rehab, Y. Siwar, and Z. Mourad. Machine learning for epilepsy: A comprehensive exploration of novel EEG and MRI techniques for seizure diagnosis. *Journal of Medical and Biological Engineering*, 3(44):317–336, 2024. https://doi.org/10.1007/s40846-024-00874-8.

[14] Antonio Quintero-Rincón, Valeria Muro, Carlos D'Giano, Jorge Prendes, and Hadj Batatia. Statistical model-based classification to detect patient-specific spike-and-wave in EEG signals. *Computers*, 9(4):1–14, 2020. https://doi.org/10.3390/computers9040085.

[15] Steven C. Chapra. *Applied Numerical Methods with MATLAB for Engineers and Scientists*. McGraw Hill, 2023.

[16] John Neter, Michael H. Kutner, Christopher J. Nachtsheim, and William Wasserman. *Applied Linear Statistical Models*. McGraw-Hill, Boston, MA, 4th edition, 1996.

[17] Samprit Chatterjee and Ali S. Hadi. *Regression Analysis by Example*. Wiley, Hoboken, NJ, 5th edition, 2012.

[18] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An Introduction to Statistical Learning with Applications in R*. Springer, New York, NY, 2013.

[19] Norman Draper and Harry Smith. *Applied Regression Analysis*. Wiley, New York, NY, 3rd edition, 1998.

[20] Andrew Gelman and Jennifer Hill. *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Cambridge University Press, Cambridge, UK, 2007.

Rule-Based Matching for Real Estate Features Detection

Mateo Agustín Ibañez Gutkin[1][0009−0008−2819−7067], Alvaro A. Pagano[1][0009−0001−1367−3218], Luciana Tanevitch[1][0000−0002−5322−9314], and Diego Torres[1,2][0000−0001−7533−0133]

[1] LIFIA-CICPBA, Facultad de Informática, UNLP. La Plata, Argentina
mateo.ibaniez@lifia.info.unlp.edu.ar
alvaro.pagano@lifia.info.unlp.edu.ar
luciana.tanevitch@lifia.info.unlp.edu.ar
diego.torres@lifia.info.unlp.edu.ar
[2] Departamento de Ciencia y Tecnología, UNQ. Bernal, Argentina

Abstract. Most of the information about real estate for sale in the Buenos Aires province, Argentina is unstructured, which means that it does not always follow the same format, making extraction a challenging process. Variability in wording, human errors, noise, and incomplete data further complicate the task. Given the large volume of information available, automated techniques are required to transform unstructured text into structured data. This article presents an approach to extract attribute-value pairs from the information contained in the property listings for the province of Buenos Aires, in order to incorporate this data into a knowledge graph. The approach uses pattern-based information extraction for 17 features with an exhaustive evaluation over two datasets: a ground truth labeled by experts and a dataset containing a real-world use case. The results demonstrates accurate values.

Keywords: Information Extraction · Rule-based matching · Natural Language Processing · Knowledge Graph Completion

1 Introduction

Most of the information available on the Web is published in natural language, making it challenging for machines to automatically process those data and draw inferences from them. Automatic techniques are required to convert text into information. To address this challenge, information extraction (IE) [3] techniques are used to automatically identify and extract structured data from unstructured sources.

Several cases, the result of the information extraction is inserted in a knowledge graph (KG)[9]. Knowledge graphs [5] are structured representations of information in the form of entities connected by relationships. Since knowledge graphs support automatic reasoning[14, 13], they can be highly effective in improving search engines and other analytical tools.

Specifically, the information related to the Real Estate market is more available and updated on the Web rather than official information. Real estate market information appears on the Web on specific sites, including a typical advertising description with a set of structured information. The description is written in natural language including information related to different relevant attributes in the description of the building. This work is focused in real estate listing published in Buenos Aires, Argentina which involves the use of Spanish language, which is not well developed in most of the NLP tools. Hence, the difficult of attributes detection in natural language is combined with the specific language making that detecting this type of attribute is currently a challenge.

Thus, the addressed problem is to extract attribute-value pairs embedded in the descriptions of real estate listings to improve a knowledge graph of real estate domain. An attribute-value pair describes a relationship between a feature (attribute) and its corresponding information (value). For example, an attribute value pair could be `{"address" : "calle 120 y 50"}`.

There are several approaches to information extraction using Natural Language Processing (NLP). A commonly used method is Rule-Based Matching (RBM), which has been applied in various contexts, including medical record analysis [4, 7]. This approach enables the identification of specific entities through predefined rules, facilitating the structuring of information extracted from unstructured documents. However, rule-based methods are not the only alternative for NLP. Supervised learning techniques, such as Named Entity Recognition (NER), have proven effective in automatically identifying key terms in texts [8, 12]. In the domain of medical document information extraction, studies like [6] have analyzed the current state of these techniques and potential future directions. Another relevant approach is the use of deep learning models. In the case of evidence-based dietary recommendations, a rule-based named-entity recognition method has been employed for knowledge extraction, as described in [2] a rule-based named-entity recognition method for knowledge extraction of evidence-based dietary recommendations. For the specific case of real estate listings, few works are published. For example, a dissertation thesis about detecting distressed real estate using NLP [10] with a english listing analysis. In Spanish, Tanevitch et. al [11] evaluate different IE approaches to extract features in real-estate descriptions.

This article presents an approach for detecting real estate features in natural language descriptions of real estate listings. The approach uses pattern-based information extraction for 17 features. In addition, this article performs an exhaustive evaluation using traditional performance metrics such as precision and recall over two datasets: a ground truth labeled by experts and a dataset containing a real-world use case. Finally, the proposed approach is compared with the one proposed by Tanevitch et. al. [11] on the same dataset, demonstrating an overall improvement in performance.

The organization of this article is detailed as follows. Section 2 introduces an overview of the workflow from getting raw data to make it structured. Next, section 3 describes the real estate features that should be extracted from the listing description. Then, section 4 has a simple explanation of the approach used for natural language processing. This is followed by the section 5, which details the process of extracting attribute-value pairs for some of the most relevant attributes. Moving on, the section 6 explains the data and metrics used to evaluate the created evaluation patterns. Section 7 highlights the metrics of the information collected. Finally, section 8 indicates conclusions and possible improvements to be applied to the project as further work.

2 Data Workflow Overview

The process of obtaining and structuring data for the real estate observatory involves to get the data and to structure it according a semantic model. Figure 2 shows a data workflow overview starting in obtaining data from the Web and to save it on a Knowledge Graph. The Knowledge Graph follows a semantic formlization in terms of an ontology. A real-estate-domain ontology [1] defines the real-estate related components (such as Real Estate), their attributes (like the address) and properties (i.a. publishedIn).

As shown in Figure 2, when data is tabular it can be directly mapped to the Knowledge graph [1]. However, in natural language descriptions, since features are embeded in the text, it is needed to apply an extra step to extract the attribute-value pairs list, and then mapping them to the KG. The contribution of this article expands on this module.

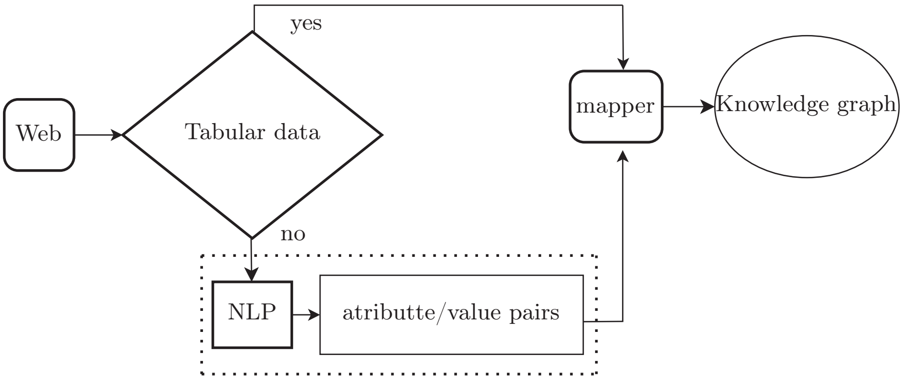


Fig. 1. Data Workflow Image

3 Real Estate Features Description

This section describes the main characteristics of the attributes to be detected. As real estate listings are published in the province of Buenos Aires, Argentina, the language in which all of this are written is in Spanish. However, the following section will detail the features in English and some cases are detailed between parenthesis with the Spanish version.

In particular, the real estate listing descriptions that are analyzed are written for different audiences. Some listings are intended for those who are going to buy a single family house, while other publications use a different vocabulary, as they are intended for more technical profiles.

Address is one of the most relevant attributes to be detected since it allows spatial localization. Four main writing formats can be generalized: mentions of a street, with the height of the lot (“Av. 19 n° 134”), mentions of 2 intersecting streets (sp *“calle San Martín y Villegas”*), mentions of the main street and between which it is located (sp *“Calle 1 entre 2 y Mitre”*), and addresses based on lots (sp *“lote 3 manzana E”*) [11]. There are other references to less precise locations that are not considered in this analysis, for example "Route 36 KM 11".

The **FOT** (in Spanish *Factor de Ocupación Total*) is a coefficient that determines the maximum buildable area of a parcel or lot. There are different types of FOT such as residential, commercial, industrial and others. In some cases the type of FOT may include awards for more building. Finally, the units of the FOT can be in square meters and its variants or in a proportion of the surface of the lot. In the advertisements it usually appears in two forms, one where it is indicated with the word “fot” or “FOT” followed by the numerical value (for example “fot 0.9”) or also in natural language of the style "FOT maximum residential of 3" (sp. *"FOT máximo residencial de 3"*).

It is necessary to identify whether the lot occupied by the property is **irregular** or not. The attribute seeks to identify land with irregular dimensions and determine a truth value. Generally, the listings includes the word “irregular”, as well as words such as "hammer", "trapezoid" or "triangular" (sp. *"martillo"*, *"trapecio"*, *"triangular"*) in reference to the shapes that the lot possesses.

The lot side **dimensions** are the front and back measurements of a lot. When lot is squared, listings usually refer to this feature (for example, sp *"10 x 50"*, *"Frente: 10, Fondo: 50"*), and may include the unit of measure. In irregular lots more sizes can be mentioned since the shape is not squared, for example "10 x 50 x 30 x 40".

Some lots are located at the intersection of two streets, meaning they are on a **corner**. The traditional ways in which this attribute appears are with the words "corner", or phrases such as "it is located in the corner". The challenges for this attribute is the appearance of references to nearby spaces, such as "police station on the corner".

The characteristic **neighborhood** makes it possible to determine in which subdivision of the locality the property is located. This characteristic is difficult to recognize, even for humans, because the word "neighborhood" is usually omitted in the description and the property is described as being within a spatial delimitation with the expression "located in".

Hence, it is also important to identify properties located in a **gated urbanization**. These are usually identified by phrases such as "closed urbanization" or "closed neighborhood". In addition, some advertisement include additional information about other types of amenities, such as "club house". The properties may belong to semi-enclosed neighborhoods or **semi-gated urbanizations**. This feature is similar to gated community, except that it includes the term "semi-closed" or similar.

The **fronts** attribute indicates the number of sides the lot has that are adjacent to any of the surrounding streets. Generally they have a single facade, but there are lots that have access from more than one street, and in those cases they will have more than one facade. The most common formats to mention the facades of a lot are "lot with 2 facades", "exits to 3 streets", "the lot has access from 3 fronts", or "lot with triple facade".

To indicate that the property has a **swimming pool**, the word "swimming-pool" is usually used. The difficulty arises when multiple listings mention the community swimming pools of a gated community or condominium. This feature usually appears with other condominium amenities such as a gym, tennis court or multipurpose room. And another less common difficulty, but one that also occurs, is the mention of canvas swimming pools, which are not considered in the detection of the feature.

The **legal status** about the possessory right of the property is analyzed to determine whether the bidder has the right to the property or not. In particular, the cases in which the legal condition of the land is of the type of usucapion, cession of possession rights or possesses a contract of sale are detected. These conditions are usually expressed with phrases containing the words "title", "rights", "usucapion" or phrases such as "does not pay expenses until possesion".

If the building referred to has not yet been built, but the project has been approved, the offer is called a **pre-sale**. The word "pre-sale" usually appears in these cases, but it is also used to refer to a medium or long term investment, such as financing, trust, installment or medium term possession. Phrases such as "development opportunity" (sp. *"posibilidad de desarrollo"*, or "building proposal" (sp. *"propuesta de construcción"*) are some examples.

The lots that cannot be subdivided are labeled under the **undivided** part regime. In the listings is mentioned that the lot is a fraction of a larger lot without legal subdivision, has an undivided part, says "undivided fraction" or says "it has an undivided deed" (sp. *"posee escritura indivisa"*). In some cases, however, the description states that they are not undivided or that the "subdivision is in process," making the task of identification more difficult.

Identifying whether the lot has a part of its structure **to demolish** is relevant. This is indicated in those advertisements where it is stated that the property has a structure that does not add value, either because it is to be repaired or destroyed, or in those ads where the structure does not add value to the sale. Terms such as "to demolish" (sp. *"a demoler"*) or "ideal for builder" (sp. *"ideal constructor"*) are used.

The **multioffer** is an attribute that identifies those advertisements in which several lots are offered in a single description. In general, they are characterized by the use of the plural in the term lots, a number that indicates several lots for sale. For example, "5 lots for sale" or "10m2 lots".

Those publications that describe an improvement that increases the value of the land, which can be due to a house, a gate, a fence, a wall, etc., are considered to meet the attribute called **monetizable**. This feature is complex because we have to make sure that this construction corresponds to the land and that it is not an intermediate project.

The **condominium lot** attribute attempts to identify the parcels that are subject to the horizontal property regime. One of the difficulties with this feature is that several notices include a denial, indicating that the lot is not subject to horizontal property.

4 Information Extraction Rule Based

Rule-based matching is a technique for extracting information from unstructured text by using predefined rules. The advantages of this method are that it is easy to understand the rules by which information is extracted, not much data is needed and the analysis does not require much computation. In general, the disadvantages are usually related to the large amount of work that must be applied to define the patterns for a given context, it is not a good approach for identifying valuable information when faced with new formats in which the information is found and it is necessary to refine the constructed patterns, so the tool is usually run several times, correcting the patterns to improve performance. One application of this technique could be done by manually defining unique patterns based on explicit syntactic rules to identify words, phrases, or syntactic structures. In particular, it is recommended to use this methodology to deal with features where we have few examples, and for well-defined and constant patterns, such as date recognition, URL formats, etc.

To carry out this strategy, we use Python's SpaCy library, which consists of three rule-based matching engines that will be explained below. The first one we will explain is the Matcher, which is able to receive sequential patterns to search within a description, asking for nouns, adjectives, prepositions, etc. in a specific order. Another tool is the PhraseMatcher, which only requests text in a specific order. Finally, the library provides the DependencyMatcher, which is able to find syntactic dependencies between words that do not have to be consecutive. This tool is very good for cases where we want to find patterns that do not follow a fixed order.

5 Matchers Description

This section describes the most interesting matchers, however the complete implementation of all the matchers is availbe at `https://github.com/cientopolis/OVS-extractor-idis`.

Regarding the detection of the address attribute, there is a Matcher structured in the four possible formats, each of which has its own set of patterns.

In order to introduce the address strategy, the following matcher example will detail one of the four cases. In particular, when an address that mentions a street between two others. In detail, the particular case that contains the names of three streets, one of which is numeric followed by the house number, for example *sp "calle 7 n° 1231, entre calle Moreno y San Martin bis"*.

One extract of the matcher is described in Listing 1.1. It shows from line 9 to 11 the detection of the street name, then lines 12 to 15 the house number. Following, lines 16 to 17 a connector, then line 18 to 20 the 18-20 the name of the first adjacent street, line 21 for a connector, and finally lines 22 to 24 detect the name of the second adjacent street.

Listing 1.1. Extract from the matcher to address detection

```
 1  CALLE_SINONIMOS = ["bv.", "bv", "avs", "avs.", "av","av,"...]
 2  CALLE_SEGMENTO = ["bis", "Bis", "BIS"]  + LETRA_MAYUSCULA
 3  NUMERO_SINONIMOS = ["numero", "numeros", "nro", ... ]
 4  ANTE_NUMERO = ["km", "al", "altura", "altura:", "alt", ... ]
 5  MEDIDAS = ["metro", "metros", "m", "ms", "mt", "mts", ...]
 6  ENTRE = ["e/", "entre", "a", "a/", "esquina", "esq", "esq."]
 7  INTERSECCION = ["y", "e", "esquina", "esq", "esq."] ...
 8  [
 9      {"LOWER": {"IN":CALLE_SINONIMOS}, "OP": "?"},
10      {"POS": "NUM"},
11      {"ORTH": {"IN": CALLE_SEGMENTO}, "OP": "?"},
12      {"LOWER": {"IN":NUMERO_SINONIMOS+ANTE_NUMERO}, "OP":"*"},
13      {"IS_PUNCT": True, "OP": "?"},
14      {"LIKE_NUM": True, "OP": "?"},
15      {"IS_PUNCT": True, "OP": "?"},
16      {"LOWER": {"IN": ENTRE}},
17      {"POS": "DET", "OP": "?"},
18      {"LOWER": {"IN":CALLE_SINONIMOS}, "OP": "?"},
19      {"POS": {"IN": ["PROPN", "NUM"]}, "OP": "{1,3}"},
20      {"ORTH": {"IN": CALLE_SEGMENTO}, "OP": "?"},
21      {"LOWER": {"IN": INTERSECCION}},
22      {"LOWER": {"IN":CALLE_SINONIMOS}, "OP": "?"},
23      {"POS": {"IN": ["PROPN", "NUM"]}, "OP": "{1,3}"},
24      {"ORTH": {"IN": CALLE_SEGMENTO}, "OP": "?"}
25  ] ...
```

The following is an example of a pattern for the address format as a lot, because this is completely different from the other three formats. In this case, it would be matched with something similar to sp. *"lote nro. 5 en manzana E"*. The fragment of the matcher can be visualized in the listing 1.2. Lines 7 to 9 detect the lot name, line 10 for a connector, then 11 to 13 detect the block.

Listing 1.2. Extract from the matcher to lot address detection

```
 1  ...
 2  NOMBRE_LOTE = ["NUM", "PROPN"]
```

```
3  SOBRE_SINONIMOS = ["en"]
4  MANZANA_SINONIMOS = ["manzana", "mz", "mz.", "mza", "mza."]
5  LETRA_MAYUSCULA = [ ... ] ...
6  [
7      {"LOWER": "lote"},
8      {"LOWER": {"IN":NUMERO_SINONIMOS}, "OP":"*"},
9      {"POS": {"IN": NOMBRE_LOTE}},
10     {"LOWER": {"IN": SOBRE_SINONIMOS}, "OP":"?"},
11     {"LOWER": {"IN": MANZANA_SINONIMOS}},
12     {"LOWER": {"IN":NUMERO_SINONIMOS}, "OP":"*"},
13     {"TEXT": {"IN": LETRA_MAYUSCULA}}
14 ] ...
```

Six patterns have been created for the FOT matcher, combining the different alternatives for its appearance. The listing 1.3 shows a fragment of the matcher for cases of the form sp. *"fot residencial de 50 m2"*. Line 6 detects the word "fot" and its variants, then line 7 detects the type of fot, line 8 detects connectors, line 9 detects the fot number, and line 10 detects unity.

Listing 1.3. Extract from the matcher to detect FOT

```
1  ...
2  TIPOS_FOT = ["residencial", "comercial", "industrial", ...]
3  FOT_SINONIMOS = ['fot', 'f.o.t','F.O.T','Fot','F.o.t','FOT:']
4  FOT_CONECTOR = ["de", ":", "es", ",", ".", "="]
5  FOT_UNIDAD = ["m2", "mts2", "metros2","metros␣cuadrados",...] ...
6  [   {'LOWER': {'IN':FOT_SINONIMOS}},
7      {"LOWER": {"IN":TIPOS_FOT}, "OP":"?"},
8      {"TEXT":{"IN":FOT_CONECTOR}, "OP":"*"},
9      {"POS": "NUM"},
10     {"LOWER":{"IN":FOT_UNIDAD}, "OP":"?"}
11 ] ...
```

To detect the value of the fronts attribute, rules are defined based on a DependencyMatcher, which has a pattern for each of the mentioned formats. The listing 1.4 shows the pattern. The matcher includes three blocks. The first block (lines 2 to 3) detects the word "exit" (sp *"salida"*). The second block (line 4 to 7) detects the any word with an oblique nominal syntactic dependency with the first block. And the third block, finds a word with a numerical dependency with the second block. In conclusion, the extract of Listing 1.4 could detect texts similar to "exit to 3 streets" (sp. *"salida a 3 calles"*), where *"salida"* is the root, *"calles"* has the oblique dependency to the root, and "3" has the numerical dependency to *"calles"*.

Listing 1.4. Extract from the matcher to detect fronts

```
1  [
2      {   "RIGHT_ID": "frentes",
3          "RIGHT_ATTRS": {"LOWER": "salida"}},
4      {   "LEFT_ID": "frentes",
5          "REL_OP": ">",
          "RIGHT_ID": "calles",
          "RIGHT_ATTRS": {"DEP": "obl"},},
    {     "LEFT_ID": "calles",
          "REL_OP": ">",
          "RIGHT_ID": "num",
          "RIGHT_ATTRS": {"DEP": "nummod"},}
],
```

6 Evaluation

This work evaluates the performance of the rule-based approach in detecting attribute-value pairs in real estate descriptions.

The evaluation of the approach will include the analysis of the performance in the detection of attributes in terms of precision, recall and f1-score, then a comparison with previous works, and finally the performance in real dataset of real estate advertisements. The patterns are applied to a massive dataset, and to measure the results in this context based on a portion of the graph. Here, the goal is to analyze what happens with unreviewed data.

6.1 Metrics

In order to analyze the performance of the approach, precision, recall, f1-score metrics and accuracy are used.

Precision is calculated using Equation 1, where TP represents the pairs correctly recognized, and FP refers to cases where the model identified attribute-value pairs that were not actually present.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{1}$$

Precision allows us to assess the model's ability to correctly identify attribute-value pairs.

Recall measures the model's ability to identify all correct pairs and is computed according to Equation 2, where TP represents the pairs correctly recognized by the tool, and FN refers to mentions that should have been recognized but were not.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{2}$$

The f1-score is a harmonic measure that combines precision and recall. calculated using Equation 3 and is

$$\text{F1-score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

Finally, accuracy is the amount of correct elements detected overall. This is particularly useful in the evaluation of the real-case OVS dataset since the records were randomly selected and there are features that may contain only true negative occurrences, so this metric would be the only representative one in this cases. The formula is given in the equation **??**.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{4}$$

6.2 Datasets

Ground truth The ground truth is comma separated files with 321 tuples. Each tuple contain the actual listing text of a real estate advertising and the values of the 17 attributes for each description. The dataset was manually tagged by experts in the study of real state.

Real-case OVS The real case dataset is a comma separated file that contains 320,000 real estate advertising descriptions without any type of tabular data. This file represents a real-world case in the domain, as it was constructed using automated techniques. As a result, the information may be incomplete or unnormalized, and variable distribution could be uneven.

7 Results

In order to evaluate the approach, two evaluation steps were done. The first analyze the approach with the ground truth dataset and also compares the resulting performance with the one introduced by Tanevitch et al.[11]. The second evaluation uses the Real case OVS dataset, taking as a first step the occurrences of each variable across the 320,000 tuples, and then extracting 100 random tuples to rerun the approach. A manual analysis of the results was then performed for each attribute to calculate the various metrics.

7.1 Ground truth results

Table 1 resumes the values obtained for each metric comparing the performance of this article contribution with Tanevitch et al.[11] approach. The first column details the analyzed attribute. Then, there are two blocks of columns. Each block includes the values of metrics precision, recall and F1-Score. Note that in the Tanevitch et al. approach, patterns were designed for only 8 of the 17 features in question, which is why the metrics are "n/a" in several cases.

Almost all metrics were improved in the approach proposed in this contribution. And, none of the metrics decreased the value. The most significantly improvements were related to attributes address, fronts and dimensions. Address precision improved 0.75 points, recall improved 0.13 points, and consequently f1-score improved 0.38 points. Fronts precision improved 0.07 points, recall improved 0.52 and F1-score improved 0.4. And, dimensions precision improved 0.02, recall improved improved 0.26 points and F1-score 0.17. Most of the improvements are related to the extensive of the cases in the definition of the matchers. For example, some matchers, as address extends the number of cases that Tanevitch et al. included in their approach.

Table 1. Metrics comparison: Tanevitch et al. vs actual contribution

Attribute	Tanevitch et al.			Actual contribution		
	Precision	Recall	F1 Score	Precision	Recall	F1 Score
Address	0.24	0.59	0.34	0.78	0.72	0.72
FOT	0.85	0.92	0.89	1.00	0.96	0.98
Irregular	1.00	0.70	0.83	1.00	0.90	0.94
Dimensions	0.85	0.54	0.66	0.87	0.80	0.83
Corner	0.90	0.92	0.91	0.90	0.92	0.91
Neighborhood	0.45	0.26	0.33	0.54	0.43	0.48
Fronts	0.88	0.38	0.53	0.95	0.90	0.93
Swimming Pool	0.76	0.97	0.85	0.82	0.97	0.89
Gated Urbanization	n/a	n/a	n/a	0.97	0.97	0.97
Legal Status	n/a	n/a	n/a	0.94	1.00	0.97
Semi-gated Urbanization	n/a	n/a	n/a	1.00	0.94	0.97
Pre-sale	n/a	n/a	n/a	0.94	0.94	0.94
Undivided	n/a	n/a	n/a	0.88	1.00	0.94
To Demolish	n/a	n/a	n/a	0.97	0.97	0.97
Multioffer	n/a	n/a	n/a	0.79	0.97	0.87
Monetizable	n/a	n/a	n/a	0.98	0.96	0.97

7.2 Real-estate OVS results

As a first step, the proposed approach was executed on the 320,000 tuples of the complete dataset. The goal of this execution is to count the occurrences and then, analyze in a portion of the cases the accuracy of the approach. Table 2, has two columns, the first column has the attributes and, the second column the number of detected occurrences of each attribute. As can be seen, the attribute monetizable had the most number of occurences of 275,510 occurences. Then, it decrease the amount of occurrences with the attributes swimming-pool, dimensions and address. Swimming-pool had 68,144 occurrences. Dimensions had 64,911 occurrences, address had 67,457 occurrences. For the rest of the attributes, the value decrease and the less appearance was for legal status with only 270 occurences.

Second, 100 random cases present in the Real-estate OVS dataset were selected for a manual analysis in order to define precision, recall, f1-score and accuracy. The manual process analyze each description in natural language and check if the attribute appears and how the approach of this article detect or note the value of the attribute.

The results are shown in table 3. The first column contains the attributes, where the 17 defined attributes are mentioned, followed by the metrics to be used. It is important to note that some features may not be covered in the

Table 2. Number of matches detected on real-case ovs dataset

Attribute	Occurrences	Attribute	Occurrences
Address	67457	Gated Urbanization	19450
FOT	5728	Legal Status	270
Irregular	2159	Semi-gated Urbanization	1152
Dimensions	64911	Pre-sale	738
Corner	18905	Undivided	395
Neighborhood	35437	To Demolish	25857
Fronts	2350	Multioffer	3318
Swimming Pool	68144	Monetizable	275510
Condominium Lot	339		

ground truth due to their rarity in the dataset, and for this reason the model fails to detect them, obtaining an f1-score of 0. However, this is not a problem, as the accuracy metric shows that the model works correctly, confirming that it has adequately not identified these features.

The results shown that corner, dimensions and monetizable are the best recognized attributes. Corner had 0.93 points of f1-score, dimensions had 0.88 points of f1-score and monetizable 0.81 points of f1-score. Several attributes are not recognized. However, none of them included false positives. Irregular had the maximum precision value but a low recall value. That means that the matcher is not taking into account several other cases about that attribute.

The results shown that the first 8 attributes did not worsen much their f1-score with respect to the table 1, since some attributes improved (fronts and neighborhood), others remained practically the same (corner and dimensions) and others worsened (address, irregular, swimming pool). Due to their low frequency, the attributes FOT, condominium lot, semi-gated urbanization, legal status, pre-sale, undivided and to demolish were left with non occurrence (n.o.) value.

It should be noted that the multioffer and monetizable matchers acquired good metrics, even if the monetizable attribute is the one that usually has the most occurrences.

Table 3. Comparison of Precision, Recall, F1-score, and Accuracy for 17 attributes. Those attributes without occurrences are labeled with n.o.

Attribue	Precision	Recall	F1	Accuracy	Attribute	Precision	Recall	F1	Accuracy
Address	0.56	0.41	0.47	0.69	Gated Urbanization	0.60	1.00	0.75	0.98
FOT	n.o.	n.o.	n.o.	1.00	Semi-gated Urbanization	n.o.	n.o.	n.o.	1.00
Irregular	1.00	0.33	0.50	0.98	Legal Status	n.o.	n.o.	n.o.	0.99
Dimensions	0.91	0.85	**0.88**	0.91	Pre-sale	n.o.	n.o.	n.o.	1.00
Corner	0.87	1.00	**0.93**	0.98	Undivided	n.o.	n.o.	n.o.	1.00
Neighborhood	0.41	0.50	0.45	0.68	To Demolish	n.o.	n.o.	n.o.	0.97
Fronts	1.00	1.00	1.00	1.00	Multioffer	0.67	0.55	0.60	0.92
Swimming Pool	0.67	1.00	0.80	0.99	Monetizable	0.74	0.90	**0.81**	0.84
Condominium Lot	n.o.	n.o.	n.o.	1.00					

8 Conclusions and future work

This article introduces a rule-based matching approach to detect 17 real estate features. The article details an overview of the general workflow of a data analysis in the Land Value Observatory (OVS in spanish), then a feature explanation is described and it is followed by a detailed explanation of the most relevant implemented matchers.

Compared with a previous approach, the one introduced in this article shown a relevant improvement of the precision, recall and F1 metrics in all the attributes. Also, our approach increased the number of attributes detected with good accuracy. In addition, our approach was executed to analyse a real estate offer dataset of 320,000 advertisements from Buenos Aires, Argentina. The evaluation demonstrated several occurences of all the attributes distinguishing monetizable and address as the most relevant, and legal status as the lowest presence.

As further work, an extended evaluation is mandatory, in order to have more evidence about the attributes that in this article appear without occurrences. In addition, the combination of this approach with other NLP techniques to improve the different performances of the matchers could be done.

References

1. Dioguardi, F., Torres, D., Antonelli, R.L., Río, J.P.d.: Construcción de un grafo de conocimiento para un observatorio inmobiliario. In: XXVIII Congreso Argentino de Ciencias de la Computación (CACIC)(La Rioja, 3 al 6 de octubre de 2022) (2023)
2. Eftimov, T., Koroušić Seljak, B., Korošec, P.: A rule-based named-entity recognition method for knowledge extraction of evidence-based dietary recommendations. PloS one **12**(6), e0179488 (2017)
3. Grishman, R.: Information extraction. IEEE Intelligent Systems **30**(5), 8–15 (2015). https://doi.org/10.1109/MIS.2015.68
4. Han, Y., Han, W., Li, S., Wang, Z.: Attribute value extraction based on rule matching. In: Sun, X., Wang, J., Bertino, E. (eds.) Artificial Intelligence and Security. pp. 92–104. Springer Singapore, Singapore (2020)
5. Hogan, A., Blomqvist, E., Cochez, M., D'amato, C., Melo, G.D., Gutierrez, C., Kirrane, S., Gayo, J.E.L., Navigli, R., Neumaier, S., Ngomo, A.C.N., Polleres, A., Rashid, S.M., Rula, A., Schmelzeisen, L., Sequeda, J., Staab, S., Zimmermann, A.: Knowledge graphs. ACM Comput. Surv. **54**(4) (Jul 2021). https://doi.org/10.1145/3447772, `https://doi.org/10.1145/3447772`
6. Landolsi, M.Y., Hlaoua, L., Ben Romdhane, L.: Information extraction from electronic medical documents: state of the art and future research directions. Knowledge and Information Systems **65**(2), 463–516 (2023)
7. Panda, S., Behera, V., Pradhan, A., Mohanty, A.: A rule-based information extraction system. International Journal of Innovative Technology and Exploring Engineering **8**, 1613–1617 (07 2019). https://doi.org/10.35940/ijitee.I8156.078919
8. Petrovski, P., Bizer, C.: Extracting attribute-value pairs from product specifications on the web. In: Proceedings of the International Conference on Web Intelligence. p. 558–565. WI '17, Association for Computing Machinery, New York,

NY, USA (2017). https://doi.org/10.1145/3106426.3106449, https://doi.org/10.1145/3106426.3106449

9. Rincon-Yanez, D., Senatore, S.: FAIR knowledge graph construction from text, an approach applied to fictional novels. In: TEXT2KG/MK@ ESWC. pp. 94–108
10. Sirigiri, P.: Identification of distressed real estate properties using natural language processing/machine learning (2096) (2024), https://scholarworks.lib.csusb.edu/etd/2096
11. Tanevitch, L., Fernández, A., Del Río, J.P., Torres, D.: Attribute-Value Extraction: the case of a Real Estate Observatory. Memorias de las JAIIO **10**(1), 181–194 (2024)
12. Wu, L.T., Lin, J.R., Leng, S., Li, J.L., Hu, Z.Z.: Rule-based information extraction for mechanical-electrical-plumbing-specific semantic web. Automation in Construction **135**, 104108 (2022). https://doi.org/https://doi.org/10.1016/j.autcon.2021.104108, https://www.sciencedirect.com/science/article/pii/S0926580521005598
13. Yao, X., Van Durme, B.: Information extraction over structured data: Question answering with freebase. In: Toutanova, K., Wu, H. (eds.) Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 956–966. Association for Computational Linguistics. https://doi.org/10.3115/v1/P14-1090, https://aclanthology.org/P14-1090
14. Zhang, Y., Yao, Q.: Knowledge graph reasoning with relational digraph. In: Proceedings of the ACM Web Conference 2022. pp. 912–924. ACM. https://doi.org/10.1145/3485447.3512008, https://dl.acm.org/doi/10.1145/3485447.3512008

WEEE Prediction Model Based on Neural Networks

Jussen Facuy[1] [0000-0003-1138-4823], Ariel Pasini [2] [0000-0002-4752-7112],
Elsa Estévez [3] [0000-0002-2596-4397] and Cesar Moran[4] [0000-0002-6596-9766]

[1,4] Universidad Agraria del Ecuador - Guayaquil, Guayas, Ecuador
[1,2] Instituto de Investigación en Informática III- LIDI -Facultad de Informática (UNLP) 50
esq. 120 La Plata, Buenos Aires
Centro Asociado CIC
[3]Laboratorio de Ingeniería de Software y Sistemas de Información (LISSI)
Departamento de Ciencias e Ingeniería de la Computación – UNS
Av. San Andrés 800 – Campus de Palihue - Bahía Blanca, Buenos Aires
Centro Asociado CIC
jfacuy@uagraria.edu.ec; cmoran@uagraria.edu.ec
apasini@lidi.info.unlp.edu.ar
ece@cs.uns.edu.ar

Abstract.
The need to develop intelligent and innovative solutions to reduce pollution generated by Waste Electrical and Electronic Equipment (WEEE) led to the construction of a WEEE prediction model based on neural networks. The information supporting the model is derived from data obtained through a survey, as well as historical data on WEEE generation in Ecuador. The model aims to estimate waste generation within a specific month and year. Neural network algorithms were used for the model's functionality due to their adaptability to dynamic data like the ones utilized. The development of this model considered five phases: data collection, preprocessing, model generation, model application, and verification and continuous improvement. It is concluded that the proposed model provides a detailed description of the architecture, phases, and procedures required for its operation, facilitating its understanding and subsequent implementation.

Keywords: Prediction Model, Neural Networks, WEEE

1 Introduction

In recent years, prediction models have become a constant on a global scale in disciplines such as medicine, economics, and environmental science, among others, due to their ability to anticipate events and thus enable well-reasoned and precise decision-making [1]. From a general perspective, predictive models are tools designed to forecast future behaviors and trends, aiming to anticipate specific outcomes based on historical data and current information sources such as structured surveys, transaction records, databases, and more [2].

Prediction models are widely implemented in institutions of various fields because their scope is not limited to professional applications but also extends to everyday activities [3]. In this regard, leveraging predictive models is an advantage for making informed and statistically grounded decisions [4]. Through a data collection process, prediction models can perform a detailed analysis of the available information, allowing for the anticipation of various scenarios [5].

Prediction models integrate mathematical and statistical components into their structure, requiring historical data and other information sources for their training and operation [6]. The development of these models involves specific stages, such as data collection, data cleaning, preprocessing, progressive training, and validation, requiring technical implementations like parameter tuning, feature selection, and performance evaluation [7].

In independent developments, these models are often built in environments like Python, implementing machine learning and deep learning algorithms [8]. Python integrates specific libraries and frameworks for the development of predictive models, such as Scikit-learn, TensorFlow, Keras, and PyTorch; tools that not only facilitate the implementation of machine learning algorithms but also provide resources to optimize model performance [9]. As a result, a significant number of companies and independent developers lean towards the Python environment to manage the development of predictive models, adapting to its flexibility, accessibility, and power [8].

In this context, neural network algorithms represent one of the most powerful tools for identifying complex patterns and trends [10]. These networks have an outstanding ability to automatically learn relationships in accordance with large volumes of data [11], [12]. Neural networks consist of a structure made up of interconnected layers, called neurons, categorized into: input layer, hidden layers, and output layer [13], [14]. These layers are necessary to process and transform the information integrated into the model, progressively manipulating it in order to structure patterns [15].

Prediction models based on neural networks represent an essential component in the predictive context, as their structure enables the identification of complex and non-linear relationships in large volumes of data, detecting underlying patterns and behaviors that could otherwise go unnoticed [16]. Neural networks have the ability to learn and generalize from data, constantly adapting to the presented datasets [17], [18]. These types of models are essential for addressing complex problems, considering the non-linear interactions between the structured variables [19].

Unlike other models, such as linear regression or traditional statistical methodologies, neural networks, through their layers, are capable of capturing complex non-linear relationships between input and output variables. This is due to their ability to learn hierarchical representations of the data [20], [21]. While linear regression is based on a linear relationship between variables, neural networks do not rely on such relationships, requiring more complex structures that enable the identification of a broader range of patterns [22].

The pollution generated from waste originating from electrical and electronic devices (WEEE) represents an environmental risk [23], [24]. The degradation of these components not only releases toxic substances that degrade the soil and contaminate water, but also contributes to the accumulation of waste in the ecosystem [25]. In this sense, developing a WEEE prediction model would not only make it possible to determine the amount of waste generated, but also promote a sense of awareness among the public.

2 Model Development

Considering the issue, a prediction model was developed to detect the amount of WEEE generated in the city of Guayaquil. The model is based on information gathered through a survey and historical data on WEEE generation in Ecuador. In its functionality, it allows the selection of a specific year and month for the WEEE projection, enabling the estimation of the amount of waste generated during that period.

In this context, neural network algorithms were selected for the model development due to their power and adaptability to dynamic and non-linear data, as they allow for progressively adjusting the weights during the training process. However, difficulties also arose during the implementation process, such as the need to incorporate a considerable amount of data, structuring the model parameters, configuring the activation function, and other inherent technical issues. Despite these challenges, neural networks demonstrated outstanding results in the accuracy of the predictions, as the coefficient of determination was compared with results obtained using linear regression algorithms, where neural networks performed better in fitting the predictions to the data.

The WEEE generation prediction model required multiple phases and procedures in its development, which could be complex to understand or implement. As a result, three diagrams were designed to graphically illustrate its structure and functionality: the conceptual framework, which broadly addresses the different phases of the predictive model; the conceptual and technical framework, which shows the phases with their respective inherent processes; and the technical framework, which describes the technical procedures in detail.

2.1 Conceptual Framework

In an effort to contextualize the implementation of the predictive model, the initial step is to explain, in a generalized manner, the phases that integrate the neural network, particularly in the prediction of electronic waste. In this sense, the phases described in the conceptual framework (Fig. 1) of the model are: Data Collection, Preprocessing, Model Generation, Model Application, and Verification and Continuous Improvement.

Data represents the foundation of a predictive model, as it allows for the structuring of statistical analyses and machine learning procedures, which enable the detection of patterns, trends, and other relationships between the involved variables [26]. In this sense, data collection was an essential phase in the model's development, as it gathered consistent information on WEEE generation in Guayaquil. In the context of the project, data collection was managed through a survey directed at Guayaquil residents, where dimensions related to the production, disposal, and management of WEEE were addressed.

Subsequently, the survey responses from Guayaquil residents will be stored in the records. However, this data cannot be directly implemented into the model, as it requires preprocessing to clean, filter, and standardize it into the appropriate format.

During preprocessing, the survey records were cleaned and standardized, lemmatized, redundant data was removed, and potential errors inherent in the migration process were corrected.

Once the data processing and cleaning were completed, the next step was to define the variables that would be part of the predictive model. In this regard, 15 input variables were defined, which are: YearProjection, Income, EducationLevel, ResidenceArea, RecyclingFrequency, TV_Disposed, Computer_Disposed, Batteries_Disposed, BasicMobilePhone_Disposed, VideoGameConsole_Disposed, Tablet_Disposed, SmartMobilePhone_Disposed, SmartAppliances_Disposed, HomeAutomationDevices_Disposed, and Other_Disposed. Similarly, an output variable was also structured, defined as "TotalProductsDisposed."

With the definition of variables, the next step was the structuring of the neural networks, where the algorithms were configured, and the data was split into training and testing sets, followed by the management of the model's training. Once the training process was completed, and after technical implementations, the model was able to generate projections on the generation of electrical and electronic waste in the context of Guayaquil, for specific time periods; defined in terms of year and month.

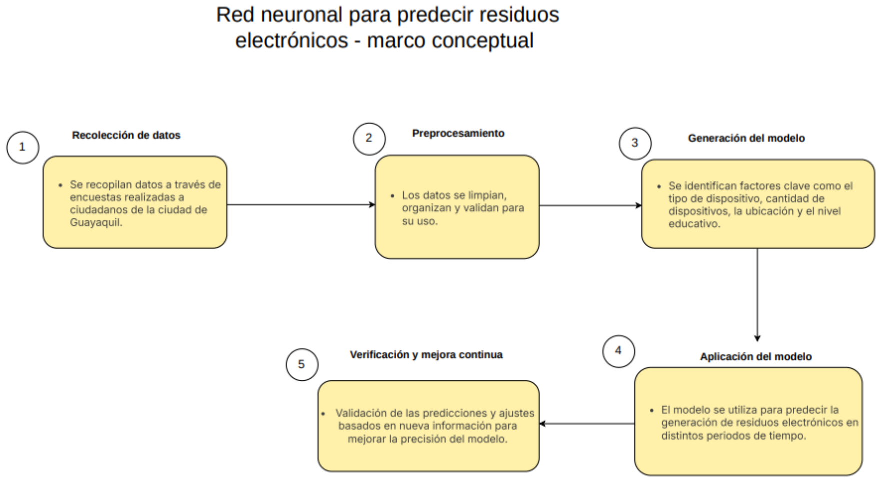

Red neuronal para predecir residuos electrónicos - marco conceptual
1
Recolección de datos
• Se recopilan datos a través de encuestas realizadas a ciudadanos de la ciudad de Guayaquil.
2
Preprocesamiento
• Los datos se limpian, organizan y validan para su uso.
3
Generación del modelo
• Se identifican factores clave como el tipo de dispositivo, cantidad de dispositivos, la ubicación y el nivel educativo.
4
Aplicación del modelo
• El modelo se utiliza para predecir la generación de residuos electrónicos en distintos periodos de tiempo.
5
Verificación y mejora continua
• Validación de las predicciones y ajustes basados en nueva información para mejorar la precisión del modelo.

Fig. 1. Conceptual Framework of the WEEE Generation Prediction Model.

Although the model, in its application, is capable of generating projections on WEEE generation in Guayaquil, this does not guarantee that the predictions will be accurate or consistent. Therefore, a process of verification and continuous improvement is required. This process not only includes statistical metrics or verification methods such as cross-validation/stratified "K-Fold," Bootstrap Sampling, or efficiency comparisons, but also takes into account emerging information in the field of WEEE generation, in order to readjust and calibrate the model, ensuring alignment with the projections.

2.2 Conceptual and Technical Framework

In the previous section, the conceptual framework was explained, where the phases that make up the prediction model for WEEE generation were discussed in a generalized manner. However, it is important to highlight that, although each phase of the conceptual framework was explained independently, they also integrate specific technical procedures that contextualize the implementation of the model. In this regard, the conceptual and technical framework (Fig. 2) serves as an intermediary between the general overview of the model and the specific details, graphically representing the structure and processes that constitute it.

In data collection, a technical process is manifested, which involves both gathering information through structured surveys, the results of which are downloaded into a CSV file, and the review and extraction of historical data, which will be necessary to complete the information. This technical procedure enabled the collection of data in an integrated manner, as it not only considered the survey results but also historical data from government research, which was essential to contextualize the incidence of WEEE in Ecuador.

In turn, the preprocessing phase integrated 3 technical procedures: data preparation, the creation of new columns, and the export of processed data. These technical procedures ensure the standardization, consistency, and quality of the data, collecting it in a structured manner that aligns with the needs of the predictive model. Similarly, the procedures made it feasible to work with cleaned and filtered data, which will reduce the margin of inconsistencies or errors in later stages.

Proceeding to the next phase, the model generation comprised 3 technical procedures: feature selection, feature loading, and data normalization. In these procedures, not only were the features that form part of the model defined and selected, but their values were also categorized and adjusted, corresponding to their specific variable type.

Subsequently, the model application continued with 4 technical procedures: data splitting, neural network design, model configuration, and model training. These procedures form the foundation of the predictive model, as they not only encompass the configurations of the neural networks but also require the processed data from the previous phases to manage the training process. This allows for the structuring of predictions regarding the generation of WEEE in the context of Guayaquil.

Finally, the process concludes with the verification and continuous improvement phase, which integrates 2 technical procedures: model validation and storing the trained model. These procedures not only ensure the efficiency, accuracy, and reliability of the predictive model but also prepare it in an appropriate format for future implementation, where it does not require retraining, but rather adjustment or updates in alignment with new relevant data in the field of WEEE management.

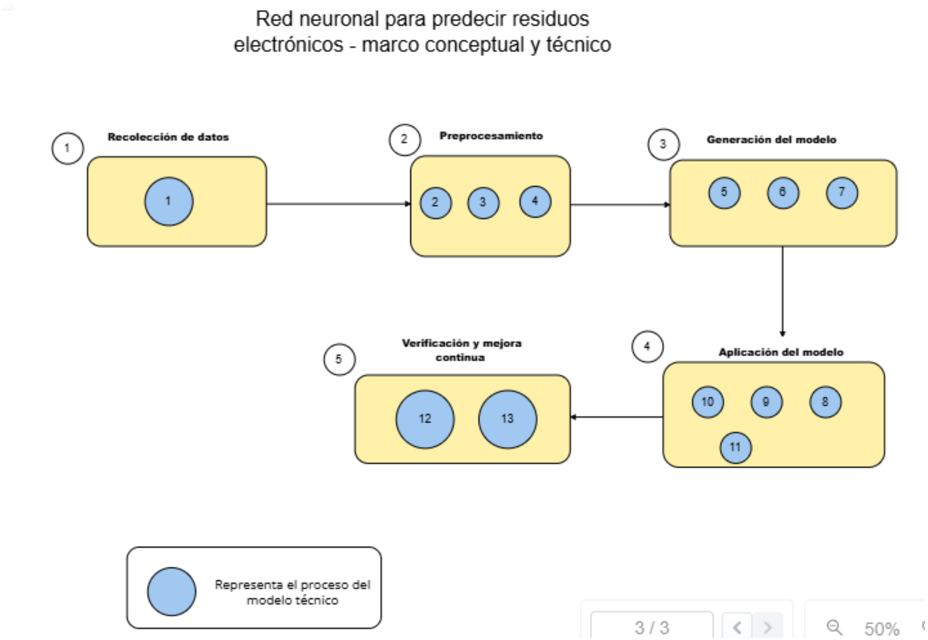

Red neuronal para predecir residuos electrónicos - marco conceptual y técnico
1
Recolección de datos
1
2
Preprocesamiento
2
3
4
3
Generación del modelo
5
6
7
4
Aplicación del modelo
10
9
8
11
5
Verificación y mejora continua
12
13
Representa el proceso del modelo técnico
3 / 3
50%

Fig. 2. Conceptual and Technical Framework of the WEEE Generation Prediction Model

2.3 Technical Framework

In a generalized manner, the phases that integrate the WEEE generation prediction model have been reviewed, along with their respective specific procedures. However, in the conceptual and technical diagram, these procedures are represented as numbers, requiring a detailed breakdown that clarifies their functionality and relationship within each phase of the predictive model. In this context, a technical framework (Fig. 3) was structured to illustrate the procedures managed during the development of the WEEE generation prediction model, where these are presented and graphically linked, allowing the workflow to be visualized.

Initially, the technical procedures begin with data collection through a Google Forms survey, where the questions address dimensions such as the area of residence in Guayaquil, average income, education level, recycling and disposal frequency, as well as the disposal of specific electrical and electronic components such as televisions, computers, phones, consoles, appliances, etc. The responses obtained from these strategic dimensions were stored in a CSV file for subsequent analysis.

It is important to remember that preprocessing involves three technical procedures: data preparation, creation of new columns, and data export. Data preparation refers to the procedure of lemmatization, removal of redundant values, and error correction. On the other hand, the creation of new columns involves structuring additional variables to enhance the model, such as the quantity of recycled and disposed products. Similarly, in the data export process, the filtered information will be stored, compressed into a CSV file, a suitable format for processing in subsequent development stages.

Subsequently, in the model generation phase, technical procedures are involved that format the preprocessed data, including variable selection, loading of input and output variables, and data normalization. Variable selection refers to the process of identifying features that align with the context of the model, such as YearProjection, ResidenceArea, Computer_Disposed, among others. In contrast to selection, loading the input and output variables represents the structured assignment of these variables, formally loading them into the model's script. Next, it is necessary to normalize the data by adjusting their values accordingly, meaning scaling them within a range that is homogeneous for all variables.

The application of the model includes data splitting, the design of the neural network layers, the configuration of the optimizer and loss function, and the neural network training process. In the data splitting phase, the preprocessed data is divided, with 80% allocated to the training set and 20% to the test set. On the other hand, neural networks need to be structured in layers that serve specific functions: the input layer, consisting of 128 neurons, which is responsible for receiving and processing the model's input variables; the hidden layers, with 64 neurons, which apply non-linear transformations to the data through ReLU activation functions; and the output layer, composed of a single neuron, which performs a linear activation that generates the model's final prediction.

The configuration of the optimizer, using "adam," is responsible for adjusting the weights of the neural network, adapting to different learning rates during training. Meanwhile, the loss function estimation is managed through Mean Squared Error (MSE), as it allows determining the difference between the generated predictions and the actual values. Finally, the training of the neural network was set to 200 epochs, as this iteration range allows the model to progressively and consistently adjust its weights, until an appropriate convergence is achieved.

Concluding the phases, the verification and continuous improvement of the model encompasses evaluation metrics and saving the trained model. Evaluation metrics refer to the validation processes used to quantify the model's efficiency, as its performance after the training phase does not necessarily indicate adequate predictive capacity, requiring further analysis to substantiate its efficiency. The evaluation metrics implemented in the model were: comparison between neural network algorithms and linear regression, statistical evaluation metrics (Mean Squared Error, Root Mean Squared Error, and Coefficient of Determination), K-Fold cross-validation, Stratified K-Fold validation, and the Bootstrap Sampling method. Additionally, a subsequent validation was considered, aimed at checking the correspondence between the generated predictions and actual values through new data collection. Finally, the trained and validated model was saved in a .h5 file for later reuse, as this format allows storing both the defined weights and the model's architecture, enabling its loading without the need for retraining.

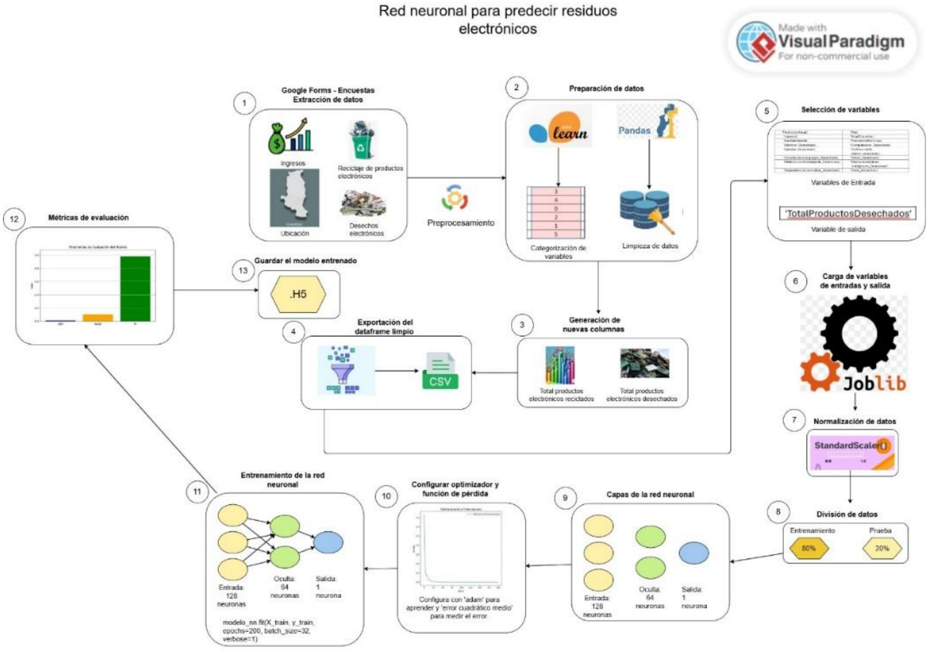

Red neuronal para predecir residuos electrónicos
Made with Visual Paradigm For non-commercial use
1
Google Forms - Encuestas Extracción de datos
Ingresos
Reciclaje de productos electrónicos
Ubicación
Desechos electrónicos
Preprocesamiento
2
Preparación de datos
learn
Pandas
Categorización de variables
Limpieza de datos
5
Selección de variables
Variables de Entrada
'TotalProductosDesechados'
Variable de salida
6
Carga de variables de entradas y salida
Joblib
7
Normalización de datos
StandardScaler()
8
División de datos
Entrenamiento
Prueba
80%
20%
9
Capas de la red neuronal
Entrada: 128 neuronas
Oculta: 64 neuronas
Salida: 1 neurona
10
Configurar optimizador y función de pérdida
Configura con 'adam' para aprender y 'error cuadrático medio' para medir el error
11
Entrenamiento de la red neuronal
Entrada: 128 neuronas
Oculta: 64 neuronas
Salida: 1 neurona
modelo_nn.fit(X_train, y_train, epochs=200, batch_size=32, verbose=1)
12
Métricas de evaluación
13
Guardar el modelo entrenado
.H5
4
Exportación del dataframe limpio
CSV
3
Generación de nuevas columnas
Total productos electrónicos reciclados
Total productos electrónicos desechados

Fig. 3. Technical Framework of the Developed Prediction Model

3 Conclusion

The previously reviewed frameworks allow for a generalized, transitional, and technical comparison of the phases and processes involved in the development of the WEEE generation prediction model. In the conceptual framework, the five phases managed in the predictive model were visualized, providing a broad explanation of their focus and relationship. Subsequently, the conceptual and technical framework presented the phases of the predictive model, each with its respective technical procedures, which were mentioned and addressed in a generalized manner. Finally, the technical framework of the model detailed the technical procedures, specifying and addressing them in their entirety.

Based on the above, the review of the frameworks not only enabled a progressive understanding of the architecture, phases, and procedures of the WEEE generation predictive model, but also successfully summarized the components covering each dimension of the process, providing a coherent visual representation and understanding of the flows and adjacent relationships between the structural factors of the predictive model, ensuring its efficiency when applied. Similarly, these representations help contextualize the macro and specific structure of the predictive model, contributing to the identification of inconsistencies or areas for improvement to consider in future implementations.

References

[1] L. Herrero-Corona, “Modelo predictivo para la selección de técnica de medición de la opinión pública,” *The Anáhuac Journal*, vol. 21, no. 2, pp. 50–77, Dec. 2021, doi: 10.36105/theanahuacjour.2021.v21n2.02.

[2] G. M. Duman and E. Kongar, “ESG Modeling and Prediction Uncertainty of Electronic Waste,” *Sustainability*, vol. 15, no. 14, p. 11281, Jul. 2023, doi: 10.3390/su151411281.

[3] S. Polymeni, E. Athanasakis, G. Spanos, K. Votis, and D. Tzovaras, “IoT-based prediction models in the environmental context: A systematic Literature Review,” *Internet of Things*, vol. 20, p. 100612, Nov. 2022, doi: 10.1016/j.iot.2022.100612.

[4] M. K. Maurya, V. K. Singh, S. K. Shaw, and M. Kumar, “Fft-asvr: an adaptive approach for accurate prediction of IoT data streams,” *J Supercomput*, vol. 80, no. 10, pp. 13976–13999, Jul. 2024, doi: 10.1007/s11227-024-05961-w.

[5] J. E. Z. Macias and S. Trilles, “Machine learning-based prediction model for battery levels in IoT devices using meteorological variables,” *Internet of Things*, vol. 25, p. 101109, Apr. 2024, doi: 10.1016/j.iot.2024.101109.

[6] N. Dalhat Mu’azu and S. Olusanya Olatunji, “K-nearest neighbor based computational intelligence and RSM predictive models for extraction of Cadmium from contaminated soil,” *Ain Shams Engineering Journal*, vol. 14, no. 4, p. 101944, Apr. 2023, doi: 10.1016/j.asej.2022.101944.

[7] S. Mao, Y. Kang, Y. Zhang, X. Xiao, and H. Zhu, “Fractional grey model based on non-singular exponential kernel and its application in the prediction of electronic waste precious metal content,” *ISA Trans*, vol. 107, pp. 12–26, Dec. 2020, doi: 10.1016/j.isatra.2020.07.023.

[8] M. Tran *et al.*, “<scp>Python-based scikit-learn</scp> machine learning models for thermal and electrical performance prediction of <scp>high-capacity</scp> lithium-ion battery,” *Int J Energy Res*, vol. 46, no. 2, pp. 786–794, Feb. 2022, doi: 10.1002/er.7202.

[9] S. D. Walton and K. R. Murphy, “Superposed epoch analysis using time-normalization: A Python tool for statistical event analysis,” *Frontiers in Astronomy and Space Sciences*, vol. 9, Nov. 2022, doi: 10.3389/fspas.2022.1000145.

[10] Y.-H. Jin, K.-H. Lee, and D.-W. Choi, “QueryNet: Querying neural networks for lightweight specialized models,” *Inf Sci (N Y)*, vol. 589, pp. 186–198, Apr. 2022, doi: 10.1016/j.ins.2021.12.097.

[11] J. Cao, D. Zhao, C. Tian, T. Jin, and F. Song, “Adopting improved Adam optimizer to train dendritic neuron model for water quality prediction,” *Mathematical Biosciences and Engineering*, vol. 20, no. 5, pp. 9489–9510, 2023, doi: 10.3934/mbe.2023417.

[12] X. Liang and J. Xu, “Biased ReLU neural networks,” *Neurocomputing*, vol. 423, pp. 71–79, Jan. 2021, doi: 10.1016/j.neucom.2020.09.050.

[13] A. Raj, G. Agarwal, P. Goyal, Y. Mittal, and S. K. Singh, “ParkSmart: Leveraging Neural Networks for Predictive Parking in Smart Cities,” in *2024 International Conference on Integrated Circuits and Communication Systems*

(ICICACS), IEEE, Feb. 2024, pp. 1–5. doi: 10.1109/ICICACS60521.2024.10499086.

[14] M. G. Vázquez Rueda, M. Ibarra Reyes, F. G. Flores García, and H. A. Moreno Casillas, “Redes neuronales aplicadas al control de riego usando instrumentación y análisis de imágenes para un micro-invernadero aplicado al cultivo de Albahaca,” *Research in Computing Science*, vol. 147, no. 5, pp. 93–103, 2018, doi: 10.13053/rcs-147-5-7.

[15] R. Zemouri, N. Omri, F. Fnaiech, N. Zerhouni, and N. Fnaiech, “A new growing pruning deep learning neural network algorithm (GP-DLNN),” *Neural Comput Appl*, vol. 32, no. 24, pp. 18143–18159, Dec. 2020, doi: 10.1007/s00521-019-04196-8.

[16] M. Gao, W. Cai, Y. Jiang, W. Hu, J. Yao, and P. Qian, “A Novel Predictive Model for Edge Computing Resource Scheduling Based on Deep Neural Network,” *Computer Modeling in Engineering & Sciences*, vol. 139, no. 1, pp. 259–277, 2024, doi: 10.32604/cmes.2023.029015.

[17] R. K. Jana, I. Ghosh, and M. W. Wallin, “Taming energy and electronic waste generation in bitcoin mining: Insights from Facebook prophet and deep neural network,” *Technol Forecast Soc Change*, vol. 178, p. 121584, May 2022, doi: 10.1016/j.techfore.2022.121584.

[18] C. J. Latha, K. Kalaiselvi, S. Ramanarayan, R. Srivel, S. Vani, and T. V. M. Sairam, “Dynamic convolutional neural network based <scp>e-waste</scp> management and optimized collection planning,” *Concurr Comput*, vol. 34, no. 17, Aug. 2022, doi: 10.1002/cpe.6941.

[19] Y. Cao, C. Gao, and Z. Yang, “A New Method of Different Neural Network Depth and Feature Map Size on Remote Sensing Small Target Detection,” 2021. doi: 10.4114/intartif.

[20] I. M. Hidalgo-Cajo, S. Yasaca-Pucuna, B. G. Hidalgo-Cajo, D. P. Hidalgo-Cajo, and N. B. Latorre-Benalcázar, “Estudio comparativo de los algoritmos backpropagation (bp) y multiple linear regression (mlr) a través del análisis estadístico de datos aplicado a redes neuronales artificiales,” *Revista Boletín Redipe*, vol. 9, no. 3, pp. 144–152, 2020, doi: 10.36260/rbr.v9i3.939.

[21] S. Chen, “Review on Supervised and Unsupervised Learning Techniques for Electrical Power Systems: Algorithms and Applications,” *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 16, no. 11, pp. 1487–1499, Nov. 2021, doi: 10.1002/tee.23452.

[22] P. K. Sarswat, R. S. Singh, and S. V. S. H. Pathapati, “Real time electronic-waste classification algorithms using the computer vision based on Convolutional Neural Network (CNN): Enhanced environmental incentives,” *Resour Conserv Recycl*, vol. 207, p. 107651, Aug. 2024, doi: 10.1016/j.resconrec.2024.107651.

[23] W. A. Bagwan, “Electronic waste (E-waste) generation and management scenario of India, and ARIMA forecasting of E-waste processing capacity of Maharashtra state till 2030,” *Waste Management Bulletin*, vol. 1, no. 4, pp. 41–51, Mar. 2024, doi: 10.1016/j.wmb.2023.08.002.

[24] A. Vishwakarma, K. Kanaujia, and S. Hait, “Global scenario of E-waste generation: trends and future predictions,” in *Global E-Waste Management Strategies and Future Implications*, Elsevier, 2023, pp. 13–30. doi: 10.1016/B978-0-323-99919-9.00013-1.

[25] J. C. Muyulema Allaica, I. D. R. Balón Ramos, I. E. Rodríguez Cortez, and F. X. Aguirre Flores, “Impacto de la Logística Inversa en la Sostenibilidad Ambiental: Una Propuesta de Marco sobre la Gestión de Residuos de Aparatos Electrónicos y Eléctricos,” *Arandu UTIC*, vol. 11, no. 2, pp. 2742–2767, Dec. 2024, doi: 10.69639/arandu.v11i2.462.

[26] H. Aly, A. Al-Ali, A. Al-Ali, and Q. Malluhi, “Analysis of Predictive Models for Revealing Household Characteristics using Smart Grid Data,” in *2023 IEEE PES Innovative Smart Grid Technologies Europe (ISGT EUROPE)*, IEEE, Oct. 2023, pp. 1–5. doi: 10.1109/ISGTEUROPE56780.2023.10407215.

Short papers

Automated and Secure Login in ALERTAR, a Resilient Cloud–Fog–Edge mHealth System for Hospital Environments

Claudio Zanellato[0009-0006-6002-8787], Rodrigo Cañibano[0000-0001-6992-5421] and Javier Balladini[0000-0002-9769-7830]

Universidad Nacional del Comahue
{claudio.zanellato,rcanibano,javier.balladini}@fi.uncoma.edu.ar

Abstract. ALERTAR system aims to assist healthcare providers in identifying early clinical deterioration in hospitalized patients on general wards. Its cloud-fog-edge architecture uses only mobile devices at the fog and edge levels to simplify its operability. As a critical healthcare system, it is of utmost importance to provide resilience across multiple layers of the architecture. To tolerate faults, the system can dynamically migrate devices between the fog and edge layers. This work focuses on describing an automated and secure login method for this system. To reduce user intervention and simplify system use, the login process has been automated by storing session credentials and utilizing a fog-level device discovery process. The design of the login mechanism is aligned with the strict security requirements of the system, considering the sensitivity of the data and the criticality of the service. This is a work in progress so performance evaluations are still pending.

Keywords: Early Warning System, Cloud-Fog-Edge, Resilient mHealth System

1. Introduction

Early Warning Systems (EWS), such as the National Early Warning Score (NEWS) 2 [1], aim to assist healthcare providers in identifying early clinical deterioration in hospitalized patients on general wards [2]. EWSs classify patients into different levels of severity or risk of developing a condition, based on clinical information such as comorbidities, vital signs, and level of consciousness. The EWS specifies the parameters of interest, and a score is calculated from the values measured in each patient. Finally, the score is used in a scale defined by the EWS that indicates the patient's risk classification. Every so often (usually within hours), a patient's new score is recalculated to understand their progress. EWSs can improve the quality of care and contribute to reducing unexpected mortality by increasing the frequency of nursing check-ups in more seriously ill patients and reducing them in less seriously ill patients [3, 4].

While EWS can be implemented through manual calculations, it increases workload and is prone to errors. If an Electronic Health Record (EHR) system is available, EWS could be integrated into it. However, conventional EHR systems, whose primary purpose is documentation, may not be suitable for effectively implementing a solution to support an improved patient care process based on EWS results. Even if its implementation were possible, the solution would not be portable to other EHR system. Our solution, the ALERTAR system, is a standalone system designed to guide and direct a care process centered on the patient, their disease, and the dynamic outcome of the EWS. ALERTAR is complementary to a EHR system and interoperable with them through the Fast Healthcare Interoperability Resources (FHIR) protocol developed by Health Level Seven International (HL7). Thus, the solution could be easily adopted by any institution.

The nursing surveillance process is guided by visual and audible alerts on staff members' mobile devices, indicating proximity or delay in follow-up visits. The interval between visits is determined based on the risk level estimated by the EWS(s) associated with the patient, based on clinical data manually entered by healthcare staff or captured by sensors. Early warnings are also issued regarding changes in risk level to enable rapid response by the medical team. Physicians receive alerts on their mobile devices and, to assess the patient's progress, have access to an EHR focused on the patient's condition.

ALERTAR has some specific characteristics that make it stand out. The system must be secure and resilient against component and communication failures, since it handles sensitive data and is vital for patient care. Additionally, the only equipment required from hospitals are mobile devices. This not only simplifies system deployment and administration but also increases resilience by allowing devices to migrate from the edge to the fog layer.

Related works on similar healthcare applications do not consider resilience in their architecture [5, 6, 7]. Add-ons have been proposed to provide resilience to existing systems, although they have strong limitations. For example, in [8] a read-only solution is presented, and in [9] local data insertion into devices is allowed, although most functionality is lost because they remain isolated from each other while the system is down.

In this article, we discuss a mobile device user login mechanism that attempts to reduce user intervention while providing a high degree of security.

2. A resilient and dynamic cloud-fog-edge architecture

Figure 1 shows the system architecture, distributed across three component levels: cloud, fog, and edge. At the **cloud** level, there is a set of servers that we simply call the "Cloud." These servers maintains the primary copy of the identifying data of a hospital and its staff (see H_1 in the cloud layer), replicas of all clinical data of current patients treated in multiple hospital sectors (see S_{H1_A}, S_{H1_B}, ..., and S_{H1_Z} in the cloud layer), and main copies of historical data (see $S_{H1\_A\_historic}$, $S_{H1\_B\_historic}$, ..., and $S_{H1\_Z\_historic}$ in the cloud layer). The remaining levels only contain mobile devices such as tablets or smartphones. At the **edge** level, there are "client" devices, which guide and direct the patient care process under the supervision of a nurse or physician, and allow manual entry of clinical data. These devices store replicas of hospital-related data (see H_1 in the edge layer) and current patients' clinical data (see S_{H1_B} in the edge layer).

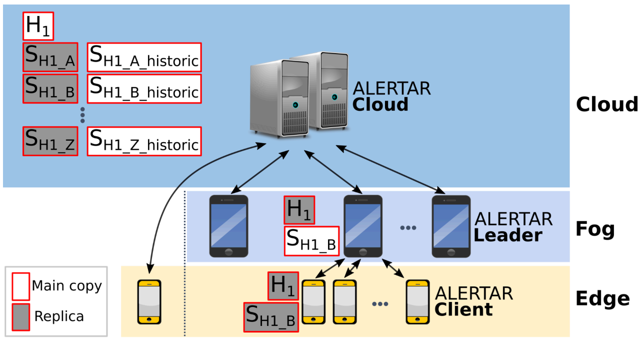


Figure 1: ALERTAR system architecture

At the **fog** layer, there is a "leader" device for each hospital sector. The "leader" device provides services to its "client" devices, processing and storing the main copy of the patients' clinical data (see S_{H1_B} in the fog layer) in a nearby infrastructure deployed using the hospital's Wi-Fi network. In turn, each "leader" device maintains a replica of the hospital's data (see H_1 in the fog layer). The leader device reduces response times and communications with the cloud compared to a traditional edge-cloud architecture. However, the architecture primarily addresses the need for resilience in two specific situations:

i. A failure affects the cloud service: client devices connected to a leader will maintain virtually the same functionality. There will be minimal degradation in terms of the inability to change the staff assigned to the hospital.

ii. A failure affects a leader's service: any client device can replace it, as they maintain replicas of all their data. We call our architecture "dynamic" because mobile devices have the ability to move from the edge to the fog levels.

3. Login mechanism

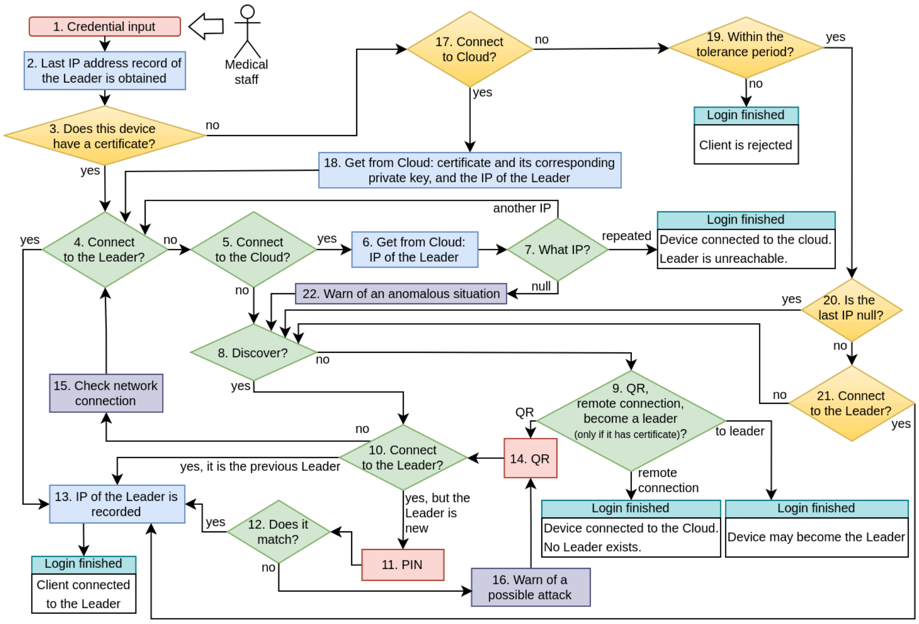

1. Credential input
Medical staff
2. Last IP address record of the Leader is obtained
3. Does this device have a certificate?
no
yes
17. Connect to Cloud?
no
yes
19. Within the tolerance period?
yes
no
Login finished
Client is rejected
18. Get from Cloud: certificate and its corresponding private key, and the IP of the Leader
another IP
yes
4. Connect to the Leader?
no
5. Connect to the Cloud?
yes
6. Get from Cloud: IP of the Leader
7. What IP?
repeated
Login finished
Device connected to the cloud. Leader is unreachable.
no
null
22. Warn of an anomalous situation
yes
20. Is the last IP null?
no
8. Discover?
no
yes
15. Check network connection
9. QR, remote connection, become a leader (only if it has certificate)?
QR
no
21. Connect to the Leader?
yes
to leader
10. Connect to the Leader?
no
yes, it is the previous Leader
14. QR
yes, but the Leader is new
remote connection
13. IP of the Leader is recorded
yes
12. Does it match?
no
11. PIN
Login finished
Device connected to the Cloud. No Leader exists.
Login finished
Device may become the Leader
Login finished
Client connected to the Leader
16. Warn of a possible attack

Figure 2: Flow diagram of the login mechanism

Figure 2 presents a flowchart of the login mechanism, which is only partially implemented. It is designed to establish a secure connection from a client device to the ALERTAR system. To provide a resilient solution, when the Cloud is unavailable, connections to the leader device are prioritized over connections to the cloud. This way, when the cloud service is unavailable, the leader device and its clients can maintain the provision of ALERTAR's core services[1] without delay or intervention.

To authenticate, both the cloud and the leader devices have a public key certificate. The cloud has a certificate issued by a public certificate authority, and the leader and client devices have a certificate issued by a private certificate authority (the cloud itself). A client device's certificate is only used when that device acts as a Leader. The cloud delivers certificates to the client devices once, upon first contact after the client app is installed (steps 3, 17, and 18 in the figure). If the cloud is inaccessible to the client, it is allowed to operate temporarily without a certificate (steps 19, 20, and 21).

The typical execution branch starts at step 4, meaning the client already has a certificate that will allow it to act as leader at any time. The client attempts to connect to the last leader device it was connected to. If possible, the process ends by authenticating the user. If contact cannot be established with the previous leader device (perhaps it is the same one but its IP address may have changed), the cloud is contacted (step 5) so that, if a leader device is currently connected to it, it can report

[1] The only service that is currently not permitted is the incorporation of new healthcare personnel into the system.

its IP address (step 6). If the reported IP address is the same as the one previously registered, and because establishing a direct connection to that IP address is impossible, the client remains connected to the system through the cloud and completes the login process (this allows remote work and also allows operation when the hospital's local area network has failed). If the IP address is different from the one registered, an attempt will be made to connect to that leader device (return to step 4). It may also happen that there is no leader connected to the cloud. This is an anomalous situation (step 22) that should be resolved because the cloud is operational.

When the client is not receiving information from the cloud about who the leader device is, a discovery process is initiated. This process attempts to resolve the connection problem transparently to the user using the Simple Service Discovery Protocol (SSDP). If the certificate presented by the leader device is not the same as that of the previous leader, the user is asked to enter a 4-digit PIN displayed on the leader device. The PIN entered on the client device must match the last 4 hexadecimal digits of the certificate's serialNumber field, thus preventing an unauthorized device from pretending to be the leader device. When the discovery process is unsuccessful, the following options are possible: (1) scanning a QR code displayed on the leader device, containing its certificate and IP address; (2) remaining connected to the cloud to retrieve potentially outdated data; or (3) converting the device into a leader.

4. Conclusions and future work

This article details a login mechanism for a resilient cloud-fog-edge architecture system based on mobile devices for use in a hospital setting. The mechanism aims to be secure and resilient to prevent user intervention as much as possible. The system has been partially implemented, and performance experiments are planned to evaluate login times and energy consumption, particularly regarding the discovery mechanism introduced in this article.

5. References

1. Royal College of Physicians. National Early Warning Score (NEWS) 2: Standardising the assessment of acute-illness severity in the NHS. Updated report of a working party. London: RCP, 2017.
2. Saab MM, McCarthy B, Andrews T, et al. The effect of adult Early Warning Systems education on nurses' knowledge, confidence and clinical performance: A systematic review. J Adv Nurs. 2017; 73: 2506–2521. https://doi.org/10.1111/jan.13322
3. Lee, J. R., Kim, E. M., Kim, S. A., & Oh, E. G.: A systematic review of early warning systems' effects on nurses' clinical performance and adverse events among deteriorating ward patients. Journal of patient safety, 16(3), e104-e113. LWW (2020).
4. Mathukia, C., Fan, W., Vadyak, K., Biege, C., & Krishnamurthy, M.: Modified Early Warning System improves patient safety and clinical outcomes in an academic community hospital. Journal of community hospital internal medicine perspectives, 5(2), 26716. Taylor & Francis (2015).
5. Equipos de Respuesta Rápida - Suite intraMed: detección oportuna del deterioro clínico en pacientes. https://intramed.mx/soluciones/equipos-de-respuesta-rapida.
6. Meditech Expanse: The Intelligent EHR. https://ehr.meditech.com/ehr-solutions/meditech-expanse.
7. Epic. Epic Systems Corporation. [2023-11-19]. https://www.epic.com/
8. Sapphire Health Helps LCMC Health Improve Clinical Resiliency with EHR Cloud Read-Only on AWS. https://aws.amazon.com/es/partners/success/lcmc-health-sapphire-health/
9. IPeople Healthcare: Downtime Product Suite. https://www.ipeople.com/products/suite.

Contributions to the modeling and simulation of an automatic package classification system to improve decision-making in line balancing.

Esteban Acosta [1], Jose Antonio de Queiroz [2] and Adriana Gaudiani [1][0000-0003-1651-0403]

[1] Universidad Nacional de General Sarmiento, Argentina
eacosta@campus.ungs.edu.ar

[2] Universidade Federal de Itajubá, Brasil

Abstract

The growth of e-commerce has led to an increase in the complexity of automatic sorting systems (Sorters), especially evident during the COVID-19 pandemic. This is reflected in the growing number of destinations, variety of products, reduced batch sizes, diversity in box dimensions, varying routes, and the need for rapid response times, among other factors. These complexities hinder decision-making in the s ystem, particularly in developing a line balancing program for package unloading lines from the Sorter. Therefore, tools that support improved decision-making are required. This publication contributes, on one hand, to the conceptual modeling of the system using the IDEF-SIM conceptual modeling technique to better understand it. On the other hand, it contributes to the construction of the simulation model using the FlexSim® software. Finally, a heuristic-based simulation optimization methodology is proposed to enhance decision-making in balancing the sorting line..

Keywords: Sorter, simulation, IDEF-SIM, optimization, load balancing

1 Background

The automatic parcel sorting system, in the context of the COVID-19 pandemic, has led to a rise in e-commerce and, consequently, an increase in the complexity of package distribution, an effect that continues to this day. The main indicators of this complexity include the number of destinations, variety of products, smaller batch sizes, box dimensions, route diversity, and faster response times among others.

These complexities, combined with the characteristics of the automatic sorting system, make key decision-making processes more difficult, such as developing a line balancing program for package unloading within the automatic sorting system [6]..

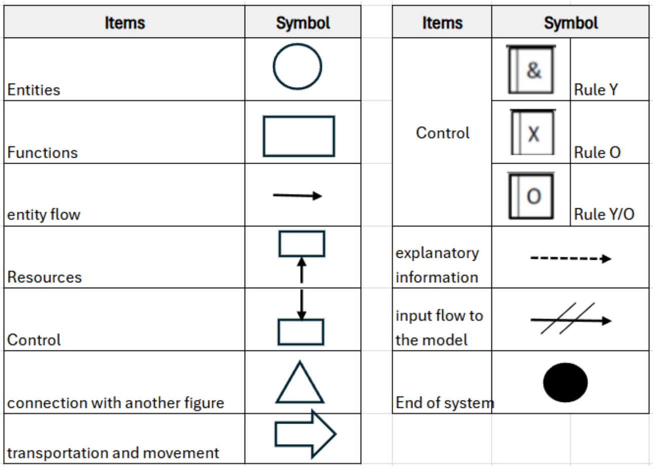
This line balancing program involves configuring the outputs of conveyor belts based on variables such as the number of destinations, number of packages per destination, box dimensions, and, given limited resources, minimizing the duration of the outbound operation. In this sense, discrete-event simulation is a technology that allows different scenarios to be addressed and multiple solutions explored until an optimal one is found [2]. However, the complexity of the variables and elements that must be considered in a Sorter system also makes conceptual simulation modeling a complex task [4].

This publication addresses the research question: What are the main elements that should be considered in the conceptual modeling of an automatic parcel sorting system that affect the efficiency of its line balancing? To answer this question, the IDEF-SIM tool is used, which was specifically designed for conceptual modeling for simulation purposes [1]. Additionally, a heuristic-based simulation optimization methodology is proposed to improve decision-making in line balancing. This methodology, previously developed and applied in a different context [5], will be adapted and used to optimize the simulator being presented.

1.1 Conceptual modeling with IDEF-SIM

The IDEF-SIM technique represents the elements of a real system, such as: entities (what moves and processes in the system), functions (workstations that operate within a given process), resources (elements that will modify or transport entities and/or that execute some function), controls (logical rules that must be used in the functions and that determine the transformation or movement of entities) and entity/movement flows (represents the direction that the entity follows within the model without undergoing modifications). Table 1, presented below, establishes the symbolism associated with the different elements detailed above [7]:

Table 1: IDEF-SIM technique element and symbol



Items	Symbol	Items	Symbol	
Entities	○	Control	&	Rule Y
Functions	▭		X	Rule O
entity flow	→		O	Rule Y/O
Resources	▭ ↑	explanatory information	- - - - - - -→	
Control	↓ ▭	input flow to the model	⇸	
connection with another figure	△	End of system	●	
transportation and movement	⇨			

1.2 Automatic package sorting system (Sorter)

Sorter systems are widely used in e-commerce and postal industries, enabling greater accuracy, capacity, additional space, and more delicate package handling. These features make them the ideal solution for successful relationships between distributors and customers. While various sorter system configurations exist, we can distinguish the basic physical design of a sorter system, a 'line configuration,' from more complex sorter systems with a 'loop configuration.' Both share a central conveyor belt and branches of conveyor belts that act as ramps, separating and sorting packages [3].

2. Development

This work is the result of modeling and simulating a Sorter system from a major logistics company in our country. A modeling process was carried out using the IDEF-SIM technique, followed by the construction of the computational model using FlexSim, and finally, a heuristic optimization method was proposed for the analysis of simulation scenarios.
In the first stage, the main elements that define this conceptual model were identified. These elements are graphically represented in a simplified Sorter system diagram (Figure 1). The key components shown in Table 2 are then identified.

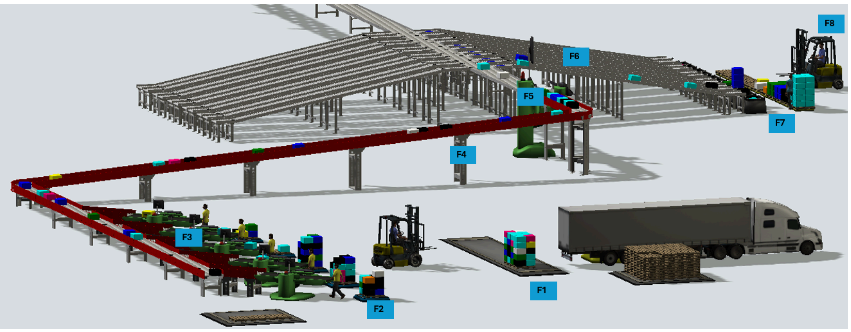
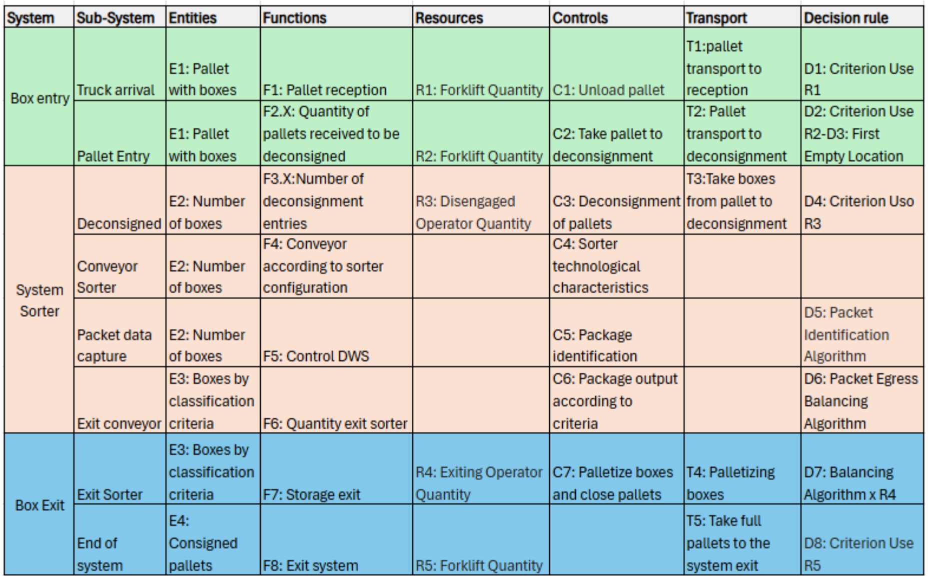


Fig. 1. Simplified Sorter System Simulated in Flexsim (source: Own elaboration)

Tabla 2: Elements of the sorter system using the IDEF-SIM technique (source: own elaboration)



System	Sub-System	Entities	Functions	Resources	Controls	Transport	Decision rule
Box entry	Truck arrival	E1: Pallet with boxes	F1: Pallet reception	R1: Forklift Quantity	C1: Unload pallet	T1:pallet transport to reception	D1: Criterion Use R1
	Pallet Entry	E1: Pallet with boxes	F2.X: Quantity of pallets received to be deconsigned	R2: Forklift Quantity	C2: Take pallet to deconsignment	T2: Pallet transport to deconsignment	D2: Criterion Use R2-D3: First Empty Location
System Sorter	Deconsigned	E2: Number of boxes	F3.X:Number of deconsignment entries	R3: Disengaged Operator Quantity	C3: Deconsignment of pallets	T3:Take boxes from pallet to deconsignment	D4: Criterion Uso R3
	Conveyor Sorter	E2: Number of boxes	F4: Conveyor according to sorter configuration		C4: Sorter technological characteristics		
	Packet data capture	E2: Number of boxes	F5: Control DWS		C5: Package identification		D5: Packet Identification Algorithm
	Exit conveyor	E3: Boxes by classification criteria	F6: Quantity exit sorter		C6: Package output according to criteria		D6: Packet Egress Balancing Algorithm
Box Exit	Exit Sorter	E3: Boxes by classification criteria	F7: Storage exit	R4: Exiting Operator Quantity	C7: Palletize boxes and close pallets	T4: Palletizing boxes	D7: Balancing Algorithm x R4
	End of system	E4: Consigned pallets	F8: Exit system	R5: Forklift Quantity		T5: Take full pallets to the system exit	D8: Criterion Use R5

For the conceptual model, the simplified sketch of a Sorter system (Figure 2) and the IDEF-SIM tool will be used as reference for its construction.
Additionally, a critical analysis is performed on the elements provided by the tool and whether any complementary components could help improve the understanding of the conceptual model. Below, in Figure 2, the main components and the construction of the IDEF-SIM diagram for the Sorter System are presented.

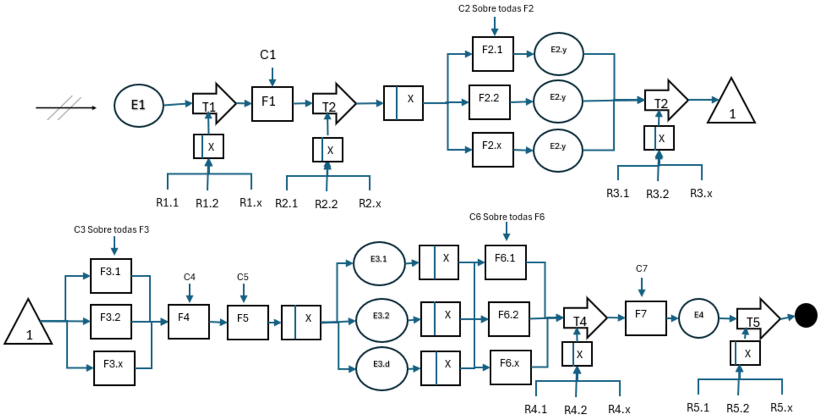


Fig. 2. Conceptual model with IDEF-SIM of a simplified Sorter system Simulated in Flexsim (source: Own elaboration)

In Figure 4, the conceptual model using IDEF-SIM is shown. Although it represents a simplification of the parcel sorting system, it was observed during the construction of the computational model that the conceptual model lacked some complementary elements useful for building the simulator. These include:

1. Complementary elements for configuring the Sorter system consisting of Functions F2, F3, and F6. It would be advisable to define the number of inputs and outputs, as this information is important for line balancing and defines system constraints.
2. Controls associated with the above points should include critical characteristics such as conveyor belt speed.
3. For identifying entities E2 and E3, it is advisable to identify the number of criteria, as they are also critical for line balancing.
4. Regarding decision rules, a complementary document specifying the algorithms used should be available, particularly for rules D5, D6, and D7.
5. It would also be of interest to identify within the IDEF-SIM model the main output variables for decision-making, such as: packages/shift, classification completion time (given a daily pallet classification plan), packages/operating cost, etc.

To verify the simulator’s potential, a proof of concept was carried out with three simulation scenarios aimed at answering: “Can one operator supply 10 inlet ports?” The simulator shows that the system collapses under that scenario and that at least

two operators are needed at the entry points, or the output distribution must be changed based on workload volume.
As explained earlier, we are working on implementing a simulator calibration methodology to improve decision-making and achieve optimal load balancing of a sorter line. This optimization is based on searching for a refined set of parameters and input variables that minimize or maximize the simulation outcomes. To develop this methodology, the five points mentioned above must be included in the conceptual model.

3. Conclusions and Future Works

We believe there is a knowledge gap regarding the application of the IDEF-SIM tool in the simulation of sorter systems. Our work provided a valuable contribution, particularly in defining complementary elements of such systems. For future work, we propose to deepen the study of these complementary elements and propose a standardization of the Sorter system using IDEF-SIM. Simulation is a valuable tool for understanding sorter systems, as it allows the analysis and optimization of sorting and unloading processes. We are working on updating the simulator with the previously indicated elements to implement the optimization methodology and assist in more efficient decision-making. Finally, we are presenting our progress in developing the simulator for sorter line balancing, and the result of the proof of concept encourages us to continue advancing.

References

[1] Acosta, E., Fernández, M., Chiodi, F., Leal, F., & Montevechi, J. Uso de la técnica IDEF-SIM en el modelado conceptual de la simulación de una línea productiva en una empresa de envases flexibles. COINI - Congreso de ingeniería Industrial, Mendoza, Argentina (2018).

[2] Binsfeld, T., & Gerlach, B. Quantifying the Benefits of Digital Supply Chain Twins—A Simulation Study in Organic Food Supply Chains. Logistics. (2022).

[3] Fikse, K. & Haneyah, S. & Schutten, Marco. Improving the performance of sorter systems by scheduling inbound containers. Journal of Medical Internet Research - J MED INTERNET RES. (2012)

[4] Gabriel, G. T., Campos, A. T., Leal, F., & Montevechi, J. A. B. Good practices and deficiencies in conceptual modelling: A systematic literature review. Journal of Simulation. (2022)

[5] Gaudiani, A, Wong, A, Luque, E, & Rexachs, D, A computational methodology applied to optimize the performance of a river model under uncertainty conditions. Journal of Supercomputing Springer (2023)

[6] Mcwilliams, Douglas L. ; Stanfield, Paul M. ; Geiger, Christopher D. The parcel hub scheduling problem: A simulation-based solution approach . Computers & industrial engineering : CAIE ; an internat. journal. - Amsterdam. (2005)

[7] Montevechi, J., Gabriel, G., Campos, A., & Leal, F. (2020). Good practices and deficiencies in conceptual modelling: A systematic literature review. Journal of Simulation.

Development of a Hand Motion Sensing Glove for Exergames: Design Evolution and Future Perspectives

Aldana Del Gener[1] [0009-0008-0024-3703] and Cecilia Sanz[1] [0000-0002-9471-0008]

Instituto de Investigación en Informática LIDI(III-LIDI) - CIC. Facultad de Informática, Universidad Nacional de La Plata, La Plata, Buenos Aires, Argentina
aldanamdg@info.unlp.edu.ar,csanz@lidi.info.unlp.edu.ar
https://weblidi.info.unlp.edu.ar/

Abstract. Exergames have gained popularity as interactive systems that promote physical activity through gaming. This paper presents the development process of a hand motion-sensing glove designed to complement an existing ankle-worn motion sensor. The glove aims to provide a more immersive exergaming experience by accurately tracking hand movements. Several prototypes were developed and iteratively improved to refine the design and functionality. This paper outlines the background of exergaming technology, reviews related work, and details the iterative development of the glove, discussing challenges encountered and improvements made. Finally, future steps for completing the final prototype are discussed.

Key words: exergames, gamepad, hand motion-sensing glove, Immersive gaming experience, videogame controller

1 Introduction

Exergames merge exercise with gaming elements to encourage physical activity in an engaging manner. These systems often rely on motion-sensing technologies to track body movements, providing real-time feedback and interaction within the game environment. While various exergaming devices exist, our focus is on enhancing hand motion tracking by developing a specialized glove that complements an already-developed ankle sensor [1]. This combination aims to provide a full-body movement tracking system, improving the user experience and broadening the application of exergames in rehabilitation, fitness, and entertainment.

This paper is organized as follows: Section 2 (Background) provides background on motion-sensing technologies in exergaming. Section 3 (Related Work) is divided into two parts: commercial solutions and academic/experimental research. This distinction highlights the gap between high-end, proprietary systems and more accessible, research-driven alternatives. Section 4 (Development Process) details the design iterations of the proposed glove, including hardware components, sensor evaluation, and challenges encountered. Section 5 discusses future improvements and presents the final conclusions.

2 Background

Motion-sensing technologies have been widely used in exergames, ranging from camera-based systems like the Microsoft Kinect to wearable sensors such as accelerometers and gyroscopes. Wearable solutions offer increased mobility and accuracy, making them ideal for real-time motion capture. In [2] the authors states wearable devices have an advantage for virtual exergames against the camera-based approaches: there is no space and place limitation. Previous research has explored the potential of sensor-equipped gloves for applications such as virtual reality (VR), rehabilitation, and gaming. However, challenges such as comfort, accuracy, latency, and durability persist in glove-based motion tracking systems.

This project aims to develop a cost-effective, standalone glove capable of seamless integration with an existing ankle sensor developed in [3] for enhanced exergaming experiences by balancing comfort, precision, and usability. Additionally, the glove is designed to recognize a series of familiar gestures used in gaming, such as mapping the X and Y axes, drawing a weapon, shooting, punching, and other game-relevant actions.

3 Related Work

3.1 Commercial Solutions

Numerous commercial products have been developed for hand-tracking and gaming applications.

One of the earliest attempts at a motion-tracking glove for gaming was the Power Glove, released in 1989 for the Nintendo Entertainment System (NES). This glove utilized bend sensors and ultrasonic technology to track hand movements, allowing players to control games with gestures. Despite its innovative approach, the Power Glove suffered from accuracy and responsiveness issues, limiting its widespread adoption. However, it remains a significant historical milestone in the development of motion-sensing gaming peripherals and serves as an early inspiration for modern hand-tracking technologies.

Nowdays, commercial products like the Leap Motion Controller (LMC) and Manus VR gloves offer high-precision tracking but often require external cameras or are cost-prohibitive. The authors of [4] analyzed the actual workspace range of the LMC and noted that tracking varies with hand orientation, finger extension, and whether the forearm is visible to the sensor. This suggests that while the LMC provides fine motion tracking, its performance is sensitive to positioning and occlusions, which can limit its applicability in dynamic or full-body exergaming contexts.

More recently, the Senso Glove developed by Senso Devices Inc.[5] offers an advanced, wireless solution for finger and hand motion tracking. It uses IMU[1]

[1] IMU (inertial measurement unit) chip contains a gyroscope and an accelerometer

sensors and provides haptic feedback via vibration motors in each finger. Designed for applications in VR and Augmented Reality (AR), it is compatible with SteamVR and popular game engines like Unity and Unreal Engine. Despite its technical advantages, the Senso Glove is relatively expensive, which makes it less accessible for general users or low-budget research projects.

3.2 Academic and Experimental Studies

Research in exergames has demonstrated the effectiveness of motion-tracking gloves in improving motor skills and engagement [6].

Recent research in gesture recognition for exergaming has focused on wearable sensor solutions combined with machine learning. In the work “A Sliding Window Approach to Natural Hand Gesture Recognition using a Custom Data Glove” [7] a custom-made glove was developed to detect natural hand gestures through bend sensors, IMUs and magnetometers. The authors employed a sliding window technique for segmenting continuous motion data and used machine learning models to recognize gestures. Importantly, the selection of gestures was grounded in prior work that analyzed common and intuitive interactions in gaming scenarios, ensuring their relevance and applicability to virtual environments. This gesture-driven design approach aligns closely with the goals of our project, as we also focus on defining gestures that are meaningful and useful within exergame contexts.

In line with the goal of affordable and efficient gesture recognition, the study titled “Automated Gestures Recognition in Exergaming” [8], explores the use of wearable sensors attached to different body parts, such as the wrist, elbow, and thigh while defining dynamic-static movement patterns to detect gestures with high accuracy using machine learning algorithms. This work highlights the potential of combining IMU-based data with intelligent classification models to automate gesture recognition and improve responsiveness in exergaming environments. This approach supports the motivation of our project, which is to enhance interaction in exergames through accessible and wearable technology.

4 Development Process

The main objective of this project is to design a glove capable of recognizing a series of predefined gestures for use in exergaming environments. This glove is intended to function in conjunction with an ankle-worn sensor—developed in previous work—to create a more immersive, full-body interaction experience in third-person video games. While the ankle sensor was tested in a first-person game environment, it presented limitations in terms of full-body interaction —particularly for upper-body gestures such as aiming, grabbing, or shooting.

To begin the development, a list of gestures was defined based on common actions in third-person games. For camera control and character movement/navigation, wrist orientation and hand posture were considered as a potential way to allow the player to rotate the in-game camera and its navigation path. Additionally, several action-based gestures were defined: a punch gesture for melee attacks, a draw weapon gesture to switch between armed and unarmed states, and a shoot gesture to simulate firing. To enable object interaction within the environment, a grabbing gesture was also specified.

The design of the motion-sensing glove underwent multiple iterations, each addressing different challenges:

4.1 Prototype 1

In this initial prototype, the gestures to be recognized by the glove were defined. A regular glove was used, and wiring was implemented to recognize events similarly to button presses. A central module, inspired by the Senso Glove, was designed and 3D printed to house the main controller (Arduino ProMicro) and the MPU9250 gyroscope, which was intended to track hand movement along the X and Y axes. Additionally, an armband was designed to contain the battery and the radio module, which would connect with the central component of the previously developed ankle sensor system [**?**]. After multiple tests, various algorithms were applied to map the X and Y axes, such as gesture classification algorithms inspired by the work of [8] aiming at recognize hand motions using IMU data. However, reproducible results could not be obtained, and the Arduino ProMicro lacked sufficient memory and processing capacity to support such computation-intensive models. Attempts to track the X and Y axes simultaneously were unsuccessful. This was due to the gyroscope's nature, which records roll, pitch, and yaw relative to a fixed (0,0,0) point, but as the hand changes position, it loses its original reference.

4.2 Prototype 2

The gyroscope was ruled out for tracking X and Y movement but remained viable for recognizing gestures such as drawing a weapon and punching. To solve X and Y tracking, a different approach was tested using two HC-SR04 ultrasonic distance sensors. These provided improved results for hand movement detection on horizontal planes. However, their accuracy depends heavily on proximity to flat surfaces such as walls, and they are susceptible to interference from nearby obstacles. Despite these limitations, this method showed promise and was retained for further refinement. Future development includes implementing compensation algorithms to reduce environmental dependency. Issues included discomfort due to rigid materials.

4.3 Prototype 3 (Current Stage)

This represents the current stage of the glove's development. In this iteration, comfort and usability were prioritized, leading to the removal of the wrist-mounted module used in previous versions. Instead, all microcontrollers and supporting electronics were mounted on the dorsal side of the hand (back of the

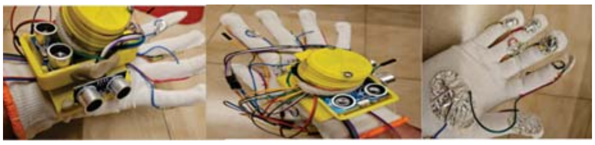


Fig. 1. Prototype 3

hand), reducing bulk and improving wearability (1). The firmware was further refined to improve axis mapping, especially for the X and Y axes during static poses, enhancing gesture stability.

Integration with the previously developed ankle sensor introduced new challenges. Specifically, during walking in place interactions, the jump phase—when the user briefly leaves the ground—caused erroneous readings in the glove. Power supply testing is still pending; for now, the glove operates while tethered to a lab power source. Final wiring for the finger sensors is also in progress, which will complete the hardware for a fully untethered prototype.

5 Conclusion and Future Work

This work presented the progressive development of an interactive glove designed to complement a previously developed ankle-based system, with the goal of providing a more immersive gaming experience by recognizing upper-body gestures. Throughout different prototype iterations, improvements were made in comfort and reading accuracy, although challenges remain when integrating the proposed glove with the ankle system.

As future work, in-depth testing will be conducted to evaluate its performance in exergame scenarios.

References

1. A. Del Gener, C. Sanz, and L. Iglesias, “Propuesta de un gamepad para sensar movimientos del jugador y su integración a un exergame,” *Revista Interacción*, vol. 4, pp. 68 – 77, 12 2023.
2. S. Ishii, M. Luimula, A. Yokokubo, and G. Lopez, “Vr dodge-ball: Application of real-time gesture detection from wearables to exergaming,” 09 2020.
3. A. Del Gener, C. Sanz, and L. Iglesias, “Exergames: propuesta de un gamepad para sensar movimientos del jugador,” 11 2022.
4. M. Tölgyessy, M. Dekan, J. Rodina, and F. Duchoň, “Analysis of the leap motion controller workspace for hri gesture applications,” *Applied Sciences*, vol. 13, p. 742, 01 2023.
5. “Senso dev center.” `https://senso.me/dev`, 2023. Accessed: 2025-04-06.
6. Ö. Dikyol, E. Çil, and T. Serif, “Enhanced therapeutic engagement: A gamified arduino glove system for hand rehabilitation,” 12 2024.
7. G. Luzhnica, J. Simon, E. Lex, and V. Pammer-Schindler, “A sliding window approach to natural hand gesture recognition using a custom data glove,” 03 2016.
8. M. Javeed and S. Aaaa, “Automated gestures recognition in exergaming,” 12 2022.

Ensuring Quality in the OECD AI Lifecycle Through ISO/IEC Standards

Juan Ignacio Torres [0000-0002-9399-7561], Ariel Pasini [0000-0002-4752-7112],
Patricia Pesado [0000-0003-0000-3482]
Institute of Research in Computer Science LIDI (III-LIDI), Faculty of Computer Science,
National University of La Plata, Argentina
{jitorres,apasini,ppesado}@lidi.info.unlp.edu.ar

Abstract. The integration of artificial intelligence (AI) technologies within organizations presents both significant opportunities and complex challenges. To manage this complexity, ISO/IEC standards provide a structured framework for the adoption and management of AI systems throughout their lifecycle. This article explores the role of ISO/IEC standards in ensuring the quality, security, and ethical alignment of AI systems, based on the lifecycle framework defined by the Organisation for Economic Co-operation and Development (OECD). The paper outlines how these standards support AI system development, from planning and design through deployment and monitoring, addressing critical issues such as governance, data quality, bias detection, and system reliability. A comprehensive quality model is proposed, drawing on ISO/IEC 25058 and 25059 standards, to assess the effectiveness and transparency of AI systems in real-world environments. The adoption of these standards is shown to enhance corporate reputation, improve regulatory compliance, and mitigate risks, positioning organizations to leverage AI technologies responsibly and efficiently.

Keywords: AI Governance, ISO/IEC Standards, AI System Lifecycle, Quality Management

1 Introduction

The adoption of artificial intelligence technologies simultaneously represents both an opportunity and a challenge for contemporary organizations. In an increasingly competitive and dynamic industrial environment, AI has become a key driver of innovation, operational efficiency, and the personalization of products and services. Its revolutionary capabilities range from process automation to the optimization of strategic decision-making, impacting sectors such as healthcare, finance, and logistics [1,2].

However, this advancement comes with technical complexities, ethical considerations, and operational challenges that may lead to issues without a structured implementation framework. The ISO/IEC standards related to AI emerge as a structured response to this need, providing organizations with methodological guidelines, best practices, and conceptual frameworks that facilitate the effective integration of intelligent systems into their operations and strategies. These standards not only clarify technical aspects but also provide tools for the organizational management of these technologies, helping companies maximize the benefits of AI [3].

In this context, the present article aims to survey quality standards applicable to the lifecycle of an AI system, according to the framework defined by the OECD [4]. Additionally, a comprehensive quality model for AI software will be proposed to establish specific criteria for measuring and ensuring the effectiveness, security, and transparency of these systems in real-world environments.

2 General Concepts

2.1 Quality Standards

ISO quality standards are international standards developed by the International Organization for Standardization (ISO) that establish guidelines and requirements to ensure that products, services, and systems meet globally recognized criteria for quality, safety, and efficiency. These standards result from international agreements among experts and cover a wide range of activities, from product manufacturing to process management and service delivery [5].

The adoption of these standards enables organizations to optimize their internal processes, reduce errors, and enhance customer satisfaction. By adhering to internationally recognized criteria, companies can improve their reputation and gain access to new markets, demonstrating their commitment to quality and excellence in their operations [6].

2.2 Artificial Intelligence

Artificial intelligence (AI) is a field of computer science focused on creating systems capable of performing tasks that typically require human intelligence, such as learning, reasoning, and perception. These systems can analyze data, recognize patterns, and make decisions with a certain degree of autonomy [7].

Artificial intelligence is based on algorithms and mathematical models that enable machines to learn from experience and adapt to new information inputs. By leveraging these technologies, computers can be trained to perform specific tasks by analyzing large volumes of data and identifying patterns within them. This has led to the development of applications across various sectors, transforming how we interact with technology in our daily lives [8,9].

3 Quality Standards in Artificial Intelligence

The ISO/IEC standards for AI provide organizations with a safer pathway for adopting intelligent technologies, reducing the uncertainty and risks inherent in the development of complex systems. Through proven methodological guidelines, these standards help identify and mitigate potential technical issues before full deployment, ensuring compliance with emerging regulations and minimizing the risk of legal and operational complications.

Implementing these standards optimizes the investment of technological and financial resources by establishing standardized processes that accelerate AI project development with a lower probability of errors. Organizations achieve better interoperability and scalability of their systems, promoting flexible designs that facilitate integration between different technologies and adaptation to changing needs while fostering the reuse of knowledge and experiences across departments.

Additionally, ISO/IEC standards strengthen corporate trust and improve relationships with stakeholders by providing transparency frameworks for the use of algorithms and data. Organizations can demonstrate their commitment to ethical and responsible AI practices, enhancing their reputation among customers, regulatory authorities, and investors while simultaneously transforming their internal operations with reliable and adaptable intelligent systems [10].

4 Lifecycle of AI Systems according to the OECD

The Organisation for Economic Co-operation and Development has defined an AI system lifecycle that consists of several key phases (Figure 1). This cycle begins with planning and design, where the system's objectives and requirements are established, considering the specific context and needs. Next, the data collection and processing phase ensures that the gathered information is relevant and of high quality for model training. Following this, the model development phase involves creating or adapting algorithms to perform the defined tasks.

Once the model is built, it enters the verification and validation phase, where its performance is assessed, and adjustments are made to ensure effectiveness and security. After passing these tests, the system moves to the deployment phase for operational use. During the operation and monitoring phase, the system's performance is continuously supervised, identifying potential improvements or necessary modifications.

It is important to note that these phases are not strictly sequential and may be iterative, depending on project needs. Furthermore, an AI system may be decommissioned at any time during the operation and monitoring phase, particularly if it fails to meet its objectives or poses unacceptable risks.

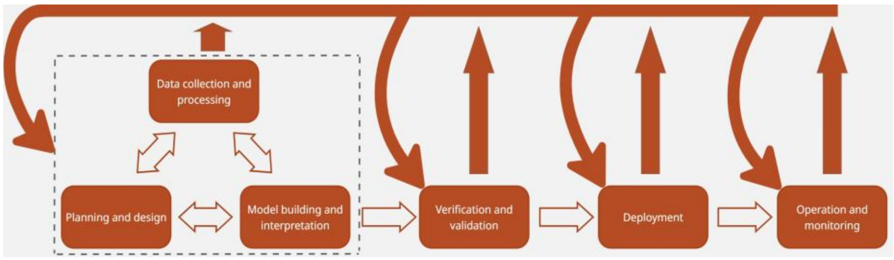

Data collection and processing
Planning and design
Model building and interpretation
Verification and validation
Deployment
Operation and monitoring

Fig. 1. AI Model Development Lifecycle according to OECD.

5 Application of Quality Standards in the OECD Lifecycle

The lifecycle of artificial intelligence systems, defined by the OECD, comprises a set of interconnected stages that require a specific regulatory framework. Each phase of intelligent system development has ISO/IEC international standards that provide precise methodological guidelines and implementation criteria.

In the initial planning and design phase, ISO/IEC 38507:2022 (Information Technology - Governance of IT — Governance implications of the use of artificial intelligence by organizations) and ISO/IEC 42001:2023 (Information Technology - Artificial intelligence — Management system) standards establish guidelines for strategic governance and objective definition, facilitating a structured framework for the conceptualization of artificial intelligence systems. During data collection and processing, ISO/IEC 5259 (Artificial intelligence — Data quality for analytics and machine learning (ML)) standards offer detailed guidance for data quality management, while ISO/IEC TR 24027:2021 (Information Technology - Artificial intelligence (AI) — Bias in AI systems and AI aided decision making) provides tools for bias detection and mitigation in data sets.

The construction and verification of the model are supported by standards such as ISO/IEC 23053 (Framework for Artificial Intelligence Systems Using Machine Learning), which outlines the architecture of machine learning systems, and ISO/IEC 25059:2023 (Software Engineering — Systems and Software Quality Requirements and Evaluation (SQuaRE) — Quality Model for AI Systems), which defines quality criteria for designing AI systems.

During the deployment and operational phases, ISO/IEC 5338:2023 (Information Technology — Artificial Intelligence — AI System Life Cycle Processes) and ISO/IEC TR 5469:2024 (Artificial Intelligence — Functional Safety and AI Systems) establish processes for operational transition and functional safety assurance, enabling continuous monitoring and the ability to decommission the system if necessary.

To evaluate the quality of artificial intelligence systems throughout these stages, an assessment model is proposed based on ISO/IEC 25058 (Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Guidance for quality evaluation of artificial intelligence (AI) systems) and 25059 standards, aligned with the OECD framework.

In the planning and design phase, the evaluation will focus on the proper application of governance principles, verifying the system’s alignment with strategic objectives and risk identification, as established by ISO/IEC 38507 and 42001.

During the data collection and processing phase, the quality and representativeness of the data used in system development will be analyzed. By applying the criteria from ISO/IEC 5259 and ISO/IEC TR 24027:2021, the accuracy, completeness, and fairness of the data will be measured, with a focus on bias detection and mitigation.

In the model construction and verification phase, the robustness and reliability of the AI system will be assessed, considering the quality criteria defined in ISO/IEC 25059. The system's architecture will be analyzed in accordance with ISO/IEC 23053, measuring its generalization capability, model explainability, and suitability for the defined purpose.

For the deployment and operation phase, the evaluation will focus on the security and stability of the system in real-world environments. The ISO/IEC 5338 and TR 5469 standards will be applied to ensure the presence of protection mechanisms against vulnerabilities and failure mitigation procedures. In addition, ISO/IEC 25059 will be used to assess the system's quality in terms of reliability and operational risk control.

Finally, in the monitoring and adjustment phase, the evaluation will continue through periodic audits and model degradation metrics. Strategies defined in ISO/IEC 25058 will be implemented to ensure the maintainability and evolution of the system, guaranteeing its reliability over time.

6 Conclusions

The adoption of AI in organizations represents a complex process that requires a systematic and structured approach. ISO/IEC standards emerge as a fundamental tool for managing this complexity, providing a regulatory framework that covers technical aspects while also addressing strategic, ethical, and operational dimensions.

The benefits of these standards go beyond the technical, becoming a competitive advantage that enhances corporate reputation, facilitates regulatory compliance, and optimizes investment in smart technologies. They reduce risks, establish standardized processes, promote interoperability, and demonstrate a commitment to ethical and transparent practices.

Throughout the article, a comprehensive model was proposed to analyze the quality of AI systems at each phase of their lifecycle. Based on ISO/IEC 25058 and 25059 standards, the model allows for a detailed assessment that spans from initial planning to continuous monitoring, ensuring risk identification, bias mitigation, and verification of the robustness of intelligent systems.

References

1. Alhosani, K., Alhashmi, S.M.: Opportunities, challenges, and benefits of AI innovation in government services: a review. Discov Artif Intell 4, 18 (2024). https://doi.org/10.1007/s44163-024-00111-w
2. Ali, O., Abdelbaki, W., Shrestha, A., Elbasi, E., Alryalat, M.A.A., Dwivedi, Y.K.: A systematic literature review of artificial intelligence in the healthcare sector: Benefits, challenges, methodologies, and functionalities. J. Innov. Knowl. 8(1), 100333 (2023). https://doi.org/10.1016/j.jik.2023.100333
3. Oviedo, J., Rodriguez, M., Trenta, A., Cannas, D., Natale, D., Piattini, M.: ISO/IEC quality standards for AI engineering. Comput. Sci. Rev. 54, 100681 (2024). https://doi.org/10.1016/j.cosrev.2024.100681
4. OECD Council: Recommendation of the Council on Artificial Intelligence. https://legalinstruments.oecd.org/en/instruments/OECD-LEGAL-0449. Last accessed 2025/03/20.
5. Bruijn, H., Duin, R., Huijbregts, M. A., Guinee, J. B., Gorree, M., Heijungs, R., Huppes, G., Kleijn, R., Koning, A., Oers, L., Sleeswijk, A. W.: Handbook on life cycle assessment: operational guide to the ISO standards. Kluwer Academic Publishers, Dordrecht, The Netherlands (2004).

6. Su, H.-C., Dhanorkar, S., Linderman, K.: A competitive advantage from the implementation timing of ISO management standards. Journal of Operations Management, 37, 31–44 (2015). https://doi.org/10.1016/j.jom.2015.03.004
7. Abbass, H.: Editorial: What is Artificial Intelligence? IEEE Transactions on Artificial Intelligence, 2(2), 94–95 (2021). https://doi.org/10.1109/TAI.2021.3096243
8. Makridakis, S.: The forthcoming Artificial Intelligence (AI) revolution: Its impact on society and firms. Futures, 90, 46–60 (2017). https://doi.org/10.1016/j.futures.2017.03.006
9. Pedro, F., Subosa, M., Rivas, A., & Valverde, P.: Artificial intelligence in education: Challenges and opportunities for sustainable development. (2019).
10. Zielke, T.: Is Artificial Intelligence Ready for Standardization? In: Yilmaz, M., Niemann, J., Clarke, P., Messnarz, R. (eds.) Systems, Software and Services Process Improvement. EuroSPI 2020. Communications in Computer and Information Science, vol. 1251, Springer, Cham (2020). https://doi.org/10.1007/978-3-030-56441-4_19

Internet deception to share IoC

Pablo Germán Maddalena Kreff[1[0009-0007-0963-4463]], Paula Venosa [1[1111-2222-3333-4444]], Patricia Bazán [1[0000-0001-6720-345X]]

[1] LINTI, Facultad de Informática UNLP

pkreff@linti.unlp.edu.ar, pvenosa@linti.unlp.edu.ar, pbaz@linti.unlp.edu.ar

Abstract. The detection of cybersecurity attacks through the collection and analysis of information is a challenge that focuses, in this work, on the use of honeypots, which are decoys that allow attacks to be studied in a controlled environment. The data collected can be used as a source of information in Cyber Threat Intelligence (CTI). Cyber Deception is a form of deception that exploits digital tools to deceive, manipulate or confuse a target, where the value lies in being attacked and investigated. Thus, honeypots constitute a Cyber Deception mechanism. Cybersecurity frameworks provide a reference model for the analysis of attacks, favoring their classification and understanding in order to mitigate them. Furthermore, these frameworks help to understand the stages of attacks linked to Cyber Deception mechanisms, including honeypots. The aim of the work is to analyze the communication mechanisms between honeypots and CTI platforms, with the aim of improving the cybersecurity strategies of organizations.

Keywords: Cyber Threat Intelligence, Honeypots, Cyber Deception, Security Orchestration Automation and Response, Indicator of Compromise, Open Source

1- Introduction

Collecting and analyzing information to detect cyber security attacks poses a challenge, as detecting malware and attacks by analyzing network traffic continues to present difficulties for those responsible for monitoring network security and managing security incidents. Although several well-known detection mechanisms exist to accurately separate malicious behavior from normal behavior, efficient detection systems are still extremely difficult.

Honeypots [1] are decoys rather than actual systems or services and it is highly likely that the traffic directed towards them is related to cyber security attacks, or discovery stage tests that are conducted for malicious purposes. Honeypots are tools that allow the study of cybersecurity attacks in controlled environments, with the objective of detecting the attack and obtaining information about the attack and the attacker, with a level of detail that other tools do not provide.

The data collected by the honeypots are the source of Cyber Threat Intelligence (CTI) information, which is the result of the enrichment of the data that is collected, processed and analyzed to understand the causes, motives, targets and attack behaviors of the threat actors[1].

This paper presents the first analyses of the possible communication mechanisms between the data collected by honeypots and the various CTI platforms, with the aim of improving organization's cybersecurity strategies.

[1] Any individual, group or organisation that carries out malicious activities with the aim of compromising the cybersecurity of devices, networks or computer systems.

2- Honeypots. Implementations and cybersecurity frameworks.

Honeypots share four key characteristics: they are deceptive, detectable, interactive and monitored. They can be defined in many ways, such as token honeypots, which mimic specific data or real data, such as documents, username and passwords, or URLs. Service honeypots mimic a specific protocol or software, such as SSH, Telnet, or an HTTP server.

Both types of honeypots can be combined to detect a specific token in a service. For example, create a username/password combination at a specific URL that certain users can detect and access, and investigate a honeypot service at the same URL to see if any username/password combination has been used.

Honeypots are deceptive and must be detectable, otherwise they cannot be attacked.

The information collected has a high value in terms of CTI, although its counterpart is the high risk that the host providing the service is actually compromised.

2.1 T-Pot

When choosing a honeypot to implement, there are several options. For this research, T-Pot[2] is chosen, being an open-source project that includes multiple honeypots, analysis, monitoring and visualization tools. T-Pot is developed and maintained by Telekom Security. The open-source community on Github[3], strongly values the project, scoring more than 7,700 stars, more than 1,200 forks, more than 200 users actively observing, and 28 contributors. As an open-source project, it has 17 releases and more than 2100 commits.

Another feature of T-Pot is that it allows honeypots to be distributed to different hosts and send the collected information to a central component. This is known as a distributed installation, with each honeypot as a Sensor, sending the data to the central Hive component where it is processed and stored.

2.2 Cyber deception and cybersecurity frameworks

Cyber Deception [2] is a form of deception that leverages digital tools to trick, manipulate or confuse a target. This collection of information from attackers helps to improve protection and mitigate them.

A honeypot is a Cyber Deception mechanism, where the value lies in being attacked and investigated.

The Cyber Kill Chain[4] framework helps to understand attacks and their phases. The model identifies the steps adversaries must complete to achieve their goal.

The relationship between the Cyber Kill Chain model and honeypots is found at several points, such as detection in the reconnaissance phase, because it allows the presence of an attacker to be identified before real assets are compromised. The information obtained from the attack, by capturing attempted delivery of malicious payloads or exploitation techniques (Exploitation phase), helps in the process of identifying and blocking the attackers's tactics.

For example, assuming the attacker attempts to install malware and establish a command and control (C2) channel, a honeypot helps to better understand the attacker's infrastructure (Command and Control phase), tools and procedures.

A honeypot that succeeds in deceiving an attacker, allows to detect the initial phases of the attack, to analyse the attacker's tactics, techniques and procedures (TTPs), to distract and divert the attacker's attention from critical systems, and thus to feed CTI information to try to interrupt, by implementing security measures, the flow of the attack.

[2] https://github.security.telekom.com/honeypot.html

[3] Data retrieved from the repository https://github.com/telekom-security/tpotce on 04/04/2025

[4] https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html

The MITRE ATT&CK[5] framework allows to classify and understand the attackers' tactics, techniques and procedures for an authentication service honeypot.

Reconnaissance tactics are achieved, including techniques such as Active Scanning. Attackers attempt to authenticate themselves to the honeypot service, which is hit by Credential Access tactics, with Brute Force techniques.

3. Interaction between threat sources and CTI

An Indicator of Compromise (IoC) is any trace or evidence that a system or network has already been attacked or compromised [3]. For the purposes of honeypots, IP addresses that originate attacks are IoCs that have a context, which is the time and destination port of the attack.

A SOAR (Security Orchestration Automation and Response) platform allows automating T-Pot's IoC lookup and publication by CTI platforms.

This is a solution so that upon receiving a honeypot attack, a CTI platform can share the IoC with other organizations, or be a source of information for itself. The CTI platform allows correlating events that have IoC so that, for example, given an IP address that sends phishing emails, it can be detected as a honeypot attacker.

A SOAR platform, in addition to performing the communication between the data collected by the honeypot and the IoC platform, can perform various actions such as blocking IP addresses, sending alerts or sending an email.

Tests were performed on Shuffle Automation[6], a widely known open-source SOAR. However, since the SOAR will be used exclusively for one-way communication from T-Pot to a CTI platform, it was replaced with a script written in Python.

Under the premise of making this connection, several issues arise, such as what information to share (source IP address, the destination port, the date and time) and how often or under what action IoCs are shared in the events to be published. For example, if you choose to share the data set related to an attack in an event, you would be sharing thousands of events per day, transmitting a 1-to-1 ratio on the CTI platform. This makes it a difficult task to manage and correlate events. If the same port is attacked from a given IP address, at different times, with different credentials, we would have an overwhelming load of events to share.

One possible approach is to share events every so often, containing a set of IoCs. For that case it remains to be solved how often is that time interval. Assuming a case where one event is shared per day, a cybersecurity analyst would have a delay (in the worst case) of almost 24 hours to get the event.

A Python script was developed to parameterize aspects such as analysis times and intervals. A balance was sought by publishing an event every one hour. In the worst case, the cybersecurity analyst would have almost a 60-minute delay between the time the attack took place and the event reaching him (assuming instantaneous transmission).

Reducing the communication time leads to more events to be published. By connecting T-Pot once every hour, we have 24 events in MISP [7](the currently implemented CTI platform). By connecting T-Pot every half hour, we double the number of events, i.e. 48. Assuming that we do not want to wait more than one minute to receive the information, we would have 1440 events, ergo we are getting closer to the initial problem.

[5] https://attack.mitre.org/

[6] https://github.com/Shuffle/Shuffle

[7] https://github.com/MISP/MISP

3.1 SOAR and CTI analyzers

Depending on the environment, needs and objectives of a cybersecurity area, a SOAR mechanism (in this case a Python script), while useful, could be handled by another CTI mechanism that allows obtaining IoCs automatically from a honeypot. That is, an Observable Analyzer[8], such as Cortex[9] or IntelOwl[10]. The observable is a certain IP address, and the goal is to enrich it. This is a crucial step in processing threat intelligence, correlating it with external sources to assess its relevance, maliciousness or association with known threats.

This relationship between T-Pot and a CTI platform does not seek to bring IoCs back to the platform every so often to be shared, but rather, once it has an IP address to query, the observables analyzer is tasked with querying the T-Pot data for their existence, if any.

These two approaches, that of a SOAR or an observables analyzer, are opposed but complementary and can work together. Table 1 presents a comparison of the two approaches, taking as parameters the access to T-Pot, the impact on its database, the latency, the action it takes and the analysis it performs on the data.

Table 1- Comparison between SOAR and an observable analyzer

	Observables Analyzers	SOAR
T-Pot Access	Requires granting direct access to the T-Pot database.	Requires granting direct access to the T-Pot database through SOAR.
Impact to T-Pot database	Query processing is in T-Pot and per observable.	The processing of the query is by SOAR, and at regular intervals.
Latency	No latency.	Latency for the frequency time at which events are shared.
Acción	Enrichment of the observable.	Sharing events and their correlation through the CTI platform.
Analysis	Active search.	Search on events previously shared and awaiting new ones through the CTI platform.

4- Results, conclusions and next steps

One result supporting this research is the implementation and collection of traffic during two months of attacks on the deployed honeypots. The months of monitoring, on an auxiliary infrastructure, have provided CTI data such as: 1- the SSH service was the most frequently attempted attack, 2- the most frequently used usernames were ‘root’ and ‘admin’, 3- the most attempted passwords were “123456” and ‘123456789’. As of the date of completion of this work, more than 182,000 attacks have been received.

The information collected from the attacks was classified by the MITRE ATT&CK framework as Active Scanning and Credential Access tactics.

The infrastructure underlying T-Pot was investigated, together with the MISP API to establish and implement a mechanism to share IoCs taken from T-Pot, under a certain time frequency.

It should be mentioned that the deployed auxiliary infrastructure has been detected and classified as ‘Honeypot’ by a well-known surface scanning service on the Internet. It is very

[8] An observable is an event (benign or malicious) in a network or system.

[9] https://github.com/TheHive-Project/Cortex

[10] https://intelowlproject.github.io/

likely that the detection is due to the number of open services, as well as the detection of mimics that do not behave identically to the original services.

As a conclusion, it is highlighted that having own CTI sources oriented to the target organization to be protected is of great value to improve the cyber security strategy.

IoCs as a source of CTI can be shared through MISP [4], within the scope of the project being carried out in the CIN Cybersecurity Commission[11] in coordination with the ARIU[12]. On the other hand, having a tool for analyzing observables, based on the data collected by T-Pot, makes it possible to enrich the active search that improves the cybersecurity strategy.

During monitoring, T-Pot's honeypots have been analyzed by various services seeking to provide information to third parties about their own surface scans on the Internet. This means that the scans received have been connections interpreted as attacks, ergo false positives.

The implementation and improvement of communication mechanisms between T-Pot and CTI platforms will continue to be monitored and investigated.

References

1. Maddalena Kreff, P. G., Gagliardi, P., Bazán, P. A., Venosa, P., Del Rio, N., Martín, S. S., & Bogado, J. (2024). Honeypots: análisis de implementaciones para ciberseguridad de una red organizacional. ISBN: 978-950-34-2428-5. Pág: 1289-1293. https://sedici.unlp.edu.ar/handle/10915/177016 .
2. Ferguson-Walter, K., Major, M., Johnson, C. and Muhleman, D. Examining the Efficacy of Decoy-based and Psychological Cyber Deception (2)(2021). https://www.usenix.org/system/files/sec21-ferguson-walter.pdf
3. NIST Special Publication 800-150 - Guide to Cyber Threat Information Sharing (10) (2016) https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-150.pdf
4. Maddalena Kreff, Pablo Germán. Malware Information Sharing Platform y su integración a CERTUNLP. Tesina de Grado. La Plata 2024. https://sedici.unlp.edu.ar/handle/10915/163504

[11] https://www.cin.edu.ar/

[12] https://riu.edu.ar/

Interpretable Machine Learning for Real Estate Valuation: A Case Study with Small Data

Emiliano Gutiérrez[1,2][0000−0002−6424−996X], Lorena Caridad López del Río[3][0000−0002−3406−9917], and Jose María Caridad Ocerín[4][0000−0003−4558−6618]

[1] Instituto de Ciencias e Ingeniería de la Computación (ICIC-UNS CONICET)
[2] Departamento de Economía. Universidad Nacional del Sur (UNS)
[3] Departamento de Economía Financiera y Dirección de Operaciones, Universidad de Sevilla.
[4] iManagement & Tourism. Sevilla

emiliano.gutierrez@uns.edu.ar, lcaridad@us.es, ccjm@uco.es

Abstract. This study explores residential property valuation in Seville, Spain, using interpretable machine learning techniques on a small dataset of 1701 sales ads of apartments collected online. Unlike conventional approaches that rely on large datasets, our research addresses the unique challenges of small data samples while maintaining model interpretability. We compare traditional hedonic linear regression with Random Forest algorithms. The results provide actionable insights for real estate stakeholders in medium-sized urban markets, bridging the gap between econometric tradition and machine learning innovation.

Keywords: interpretable machine learning · hedonic pricing · random forest,

1 Introduction

This paper examines residential property valuation in Seville, Spain - a medium-sized urban center with distinctive market characteristics. Unlike conventional housing price valuation models that typically employ large datasets, our study utilizes a small dataset, presenting unique predictive challenges that warrant methodological innovation.

From both economic and social perspectives, housing valuation represents a problem of significant importance. Consequently, our research emphasizes model interpretability across both linear regression and machine learning approaches, enabling meaningful extraction of actionable insights from our results.

The goal of this investigation is to develop machine learning models capable of identifying key determinants within Seville's real estate market. Specifically, we aim to provide interpretable results through machine learning models.

2 Motivations

A house can be considered a composite good. This type of good has a key characteristic: it can be disaggregated into distinct attributes, each of which can be individually priced being possible distinguish the implicit price of each feature.

Thus, the price of a good is revealed through the combination of its constituent characteristics. Since the seminal paper of Ridker [3] which evaluates the impact in house prices in St. Louis (United States).

Nevertheless, the relation between characteristics and prices aren't typically linear. For this reason, the use of Machine Learning algorithms emerges as a complementary alternative, capable of exploring both linear and nonlinear relationships between housing attributes and price . Machine learning methods works an alternative to a typical econometrics models. For this reason in this work we will run a random forest model with the traditional hedonic linear regression.

Machine learning approaches have primarily focused on optimizing predictive performance through data-driven modelling processes. However, the inherent opacity of many advanced algorithms ,often characterized as "black box" systems, makes the interpretability of their estimates critically important. In this context, model-agnostic interpretation techniques provide valuable insights into the prediction mechanisms of machine learning algorithms.

In this research we will use feature importance which is a technique used to evaluate the significance of each feature in a machine learning model by measuring the increase in prediction error after randomly shuffling the values of that feature [1]. A second technique for evaluate a model is the Friedman's H statistic which studies pairs of features are permuted together (to detect interactions) versus when they are permuted individually [2]. If the drop in model accuracy when shuffling two features together is greater than the sum of their individual permutation importances, it suggests a significant interaction between them.

3 Methodology and data

We developed a program to crawl one of the most important websites in Spain's real estate market: Fotocasa. We chose this option due to the possibility of obtaining a more representative sample of all transactions while simultaneously acquiring the coordinates of each property sale. The data were retrieved on 2 March 2025 and was retrieved only the apartments ads of sale.

After running the web-scraping program, the complete raw dataset was obtained in less than one second. Latitude, longitude, amenities, and physical characteristics were collected. After cleaning the data by deleting extreme values, incomplete sales ads, and incorrect information, the final number of available observations was 1701. This dataset is online in this Github repository

The second step involved locating each apartment relative to the proximity of emblematic buildings. For this purpose, we downloaded public geospatial data about hospitals, universities, subway stations, bus stops, and prisons, and estimated the nearest distance to these buildings. Thus, the final list of variables is shown in Table 1.

Table 1: List of variables

Feature	**Type**	**Description**
Price	Numerical	Price of sell published
Area	Numerical	Total area of apartment
Rooms	Numerical	Number of rooms
Bathrooms	Numerical	Number of bathrooms
Elevator	Binary	Presence of elevator (1) or absence(0)
Air Conditioning	Binary	Presence of air conditioning (1) or absence(0)
Balcony	Binary	Presence of balcony (1) or absence(0)
Terrace	Binary	Presence of terrace (1) or absence(0)
Hospital	Numerical	Distance to nearest hospital (in kilomteres)
Subway	Numerical	Distance to nearest subway station (in kilomteres)
Prison	Numerical	Distance to prison (in kilomteres)
University	Numerical	Distance to nearest university (in kilomteres)
Urban bus	Numerical	Distance to nearest urban bus stop (in kilomteres)
Intercity bus	Numerical	Distance to nearest intercity bus stop (in kilomteres)
Longitud	Numerical	Longitude
Latitude	Numerical	Latitude

In this framework, the model treats price as a function of apartment characteristics, in other words $price_i = f(x_i)$, where x_i is a vector with all the characteristics described in table 1.

4 Results

We ran a 10-fold cross-validation. In the case of hedonic linear regression, we show the coefficients of the model with the best Mean Absolute Error (MAE) in Table 2.

The results show, as expected, positive values for area, bathrooms, and amenities. Regarding proximity, for hospitals, subway, and interurban bus stations, greater distances have a positive impact. On the other hand, distance to prisons, urban bus stations, and universities shows a negative relation with price.

The geographical position was also significant at the 95% threshold. Latitude and longitude were significant but in opposite directions. In the case of latitude, north and east apartments, according to these coefficients, are in the most expensive zones.

The other method used to predict prices was Random Forests, with the optimal hyperparameter tuning: 2000 trees, the square root of the number of variables as the maximum number of features, a minimum of 1 sample per leaf, and a minimum of 2 samples to split.

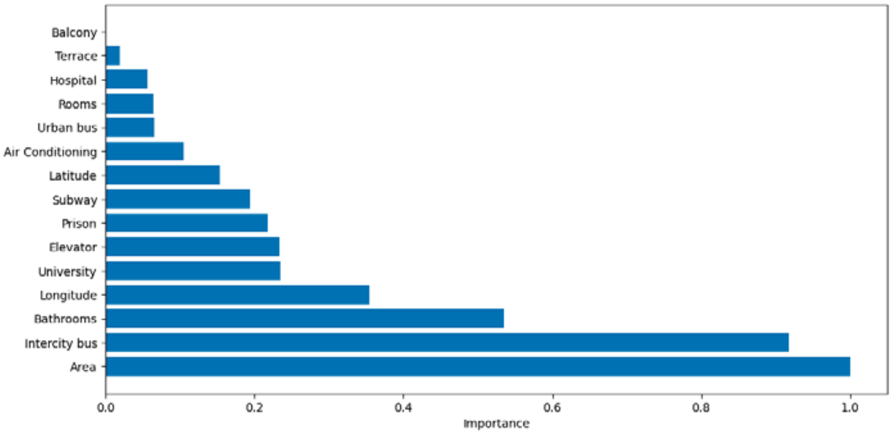
The feature importance is shown in Figure 1. It's important to notice the critical role of area in prediction. In addition, bathrooms and the proximity to the intercity bus are other variables whose relevance is important for this algorithm. Similarly, longitude, or in other words, the location between north and south is important for determining the final price of an apartment.

Table 2: Coefficients of linear hedonic regression.

	Coef.	Std.Err.	t-value	p-value
const	-38600,17	3.507,83	-11,004	0,000*
Area	1799,461	81,677	22,031	0,000*
Rooms	-12092,72	2483,27	-4,870	0,000*
Elevator	33007,35	3903,48	8,456	0,000*
Bathrooms	51270,32	4068,76	12,601	0,000*
Air Conditioning	33542,76	3832,333	8,753	0,000*
Balcony	16159,15	4925,412	3,281	0,001*
Terrace	14431,47	3886,70	3,713	0,000*
Hospital	16816,70	6523,19	2,578	0,010*
Subway	6634,66	3480,79	1,906	0,057
Prison	-26918,73	10345,72	-2,602	0,009*
University	-13695,372	3919,761	-3,494	0,000*
Intercity bus	39694,29	2862,02	13,869	0,000*
Urban bus	-14224,25	4469,89	-3,182	0,001*
Longitud	-2173811,92	1000422,18	-2,173	0,030*
Latitude	690089,62	170263,81	4,053	0,000*

R-squared: 0,741 Adj. R-squared: 0,738 No. Observations: 1531 AIC: 383,90
Df Model: 15 F-statistic: 288,8 Df Residuals: 1515 Prob (F-statistic): 0,00

Fig. 1: Feature importance



Interaction strength provides an alternative approach for examining variable importance. As shown in Figure 2, we present these interactions using Friedman's H-statistic, where Area emerges as the variable exhibiting the strongest interactions with other features. However, Figure 3 reveals that this interaction pattern is asymmetric, being particularly concentrated with bathrooms.

Neverthless examining the predictive performance between the models, random Forests is highly superior in predictive accuracy as shows 3.

Friedman's H statistic (Random forests)

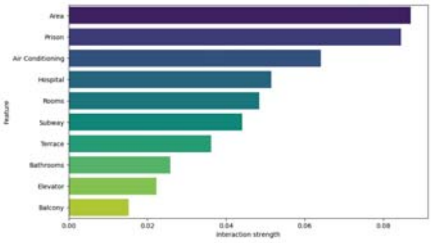


Fig. 2: General Interaction strength

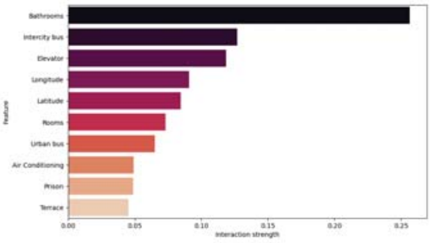


Fig. 3: Interaction with Area

Table 3: Model Performances Hedonic and Random Forest(mean of CV)

	Hedonic linear regression	**Random Forest**
RMSE	66932,59	47260,17
MAE	50886,70	32159,60
MAPE	0,36	0,20
R^2	0,74	0,87

5 Conclusions

This study contributes to the real estate market analysis of Seville by utilizing online data and machine learning algorithms. The results demonstrate the significance of key attributes such as location and square meters in determining final prices.

As future research lines, we propose examining additional interpretability techniques including Shapley values and partial dependence plots to better understand their impact on pricing. Furthermore, it would be valuable to compare these results with other machine learning algorithms to assess potential performance improvements.

References

1. Breiman, L.: Random forests. Machine learning **45**, 5–32 (2001)
2. Molnar, C.: Interpretable machine learning. Lulu. com (2020)
3. Ridker, R.G., Henning, J.A.: The determinants of residential property values with special reference to air pollution. The review of Economics and Statistics pp. 246–257 (1967)

No Latency, No Waste: How Fog Computing Optimizes Precision Agriculture

Gonçalves, Ricardo[1][0009-0005-8346-4808] Rossi, Gustavo[1,2][0000-0002-3348-2144]

[1] Universidad Abierta Interamericana. Facultad de Tecnología Informática.
Centro de Altos Estudios en Tecnología Informática - CAETI
[2] LIFIA. Facultad de Informática, UNLP
ricardo.goncalves@alumnos.uai.edu.ar,
gustavo@lifia.info.unlp.edu.ar

Abstract. The implementation of Internet of Things (IoT) technologies in precision agriculture has been revolutionizing crop resource management. Soil sensors continuously monitor moisture, temperature, and salinity, providing crucial data for precise irrigation and soil management, while weather sensors track atmospheric conditions including air temperature, humidity, precipitation, and wind speed. This data enables prediction and management of pests and diseases, while also informing harvest decisions. Continuous monitoring and advanced data analysis allow for identification of trends and anomalies, facilitating rapid and precise adjustments to agricultural operations. In this article we propose an approach to deal with the problem of latency in the use of IoT in remote areas.
.

Keywords: IoT, Latency, Fog Computing, Grafana, Monitoring, Precision Farming.

1 Introduction

Precision Agriculture, also known as Precision Farming, is an innovative and detailed approach to agricultural management that uses advanced technologies to optimize production. This methodology employs sensors, satellite imagery, drones, and GPS systems to collect field-specific data[1]. Fog Computing (also called edge cloud computing) is a service distribution paradigm for edge computing systems. It creates an intermediate layer between cloud infrastructure and network edge devices, positioned closer to endpoint equipment. This architecture plays a critical role in supporting real-time application execution and low-latency processing requirements [2]. In both scenarios, latency emerges as a critical factor. In IoT environments, latency plays an even more significant role due to the inherent nature of these systems, which often involve resource-constrained devices, wireless communication, and time-sensitive applications. This study presents an experimental framework to evaluate latency in cloud versus fog computing architectures. The paper is organized into five sessions: Session 2 establishes the motivation, Session 3 details the case study, Session 4 describes the testing methodology, and Session 5 presents the obtained results.

2 Our Approach

This work forms part of the master research carried out by Ricardo Gonçalves at UAI; we seek to address the critical challenge of latency in networks and IoT systems in precision agriculture through a proposal for the use and implementation of networks in Fog Computing. Based on architectural principles established by Bonomi et al. (2012) and later milestones of the OpenFog Consortium, the study presents an empirical evaluation that compares paradigms for the use of computing in the cloud and in the cloud in the agricultural context. To validate the research methodology, we created an experimental implementation based on AWS, where we sought to replicate one of the main operational restrictions of rural environments, which is geographic dispersion, obtained through the positioning of multi-regional AWS nodes, including humidity, temperature and lighting sensors in it. Our results demonstrated conclusive evidence that Fog nodes achieve a 70–85% reduction in end-to-end latency compared to traditional cloud processing, measured using RTT (57–140 ms vs. 300–400 ms) and one-way (3–4 ms vs. 220–225 ms) metrics. These findings not only validate the technical feasibility of Fog Computing for time-sensitive agricultural operations in resource-constrained environments, but also establish a new benchmarking methodology for evaluating edge computing performance in agricultural applications, particularly relevant for regions where reliable cloud connectivity remains an issue.

3 Case Study

We seek to use a testing environment closer to physical reality, with sensors and virtual servers. To implement the experiment, we created three servers with different roles within the cloud and fog computing architecture. The experimental setup involved deploying the following servers and devices within the Amazon AWS environment, as described below:

Server Configuration:

- **RaspBerry Server (IoT Gateway):** This server acts as an entry point for sensor data, being responsible for connecting to AWS IoT Core and sending information to Fog and Cloud servers;
- **Fog Server:** Implemented as an instance on AWS, this server processes data locally before sending it to the cloud, which reduces latency and overhead on the cloud infrastructure;

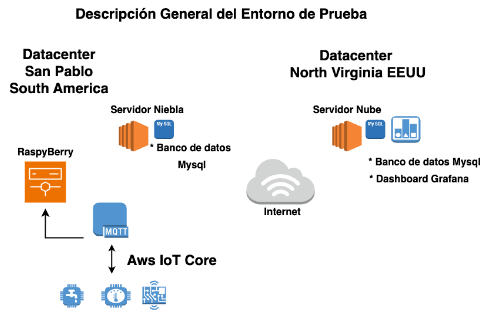
- **Cloud Server:** Instantiate on AWS with a centralized database, where collected and processed data is stored for subsequent analysis;

Aws IoT Core Virtual Sensors:

We create virtual sensors in AWS IoT Core to simulate physical sensors in a real environment. These sensors generate temperature, humidity and luminosity data, which are sent to RaspBerry via the MQTT protocol. The configuration consists of:

- Configuration of MQTT "topics" for each sensor;
- Creation of digital certificate for devices;
- Configuration of security policies to allow communication between devices and AWS;
- Configuration of an MQTT client on RaspBerry to receive and resend data.

Test environment:



4 Test Development

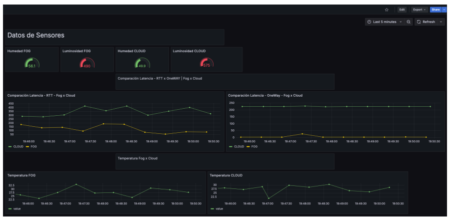
We developed six scripts, all storing data in a database. This storage enables data comparison and report generation, with results being sent to Grafana. All scripts can read sensors in AWS IoT Core: (Sensor_Humidity, Sensor_Temperature, Sensor_Light). Some scripts use Sensor_Temperature while others use Sensor_Humidity, depending on requirements. The scripts take measurements every 20 seconds and record data in the databases, and this interval is established to balance continuously updated measurements without overloading the storage and processing systems and being able to generate graphs that can be analyzed. Communication between the virtual sensors and the Raspberry server is established via the MQTT protocol, using AWS IoT Core as the message broker. To ensure secure data transmission, a set of cryptographic keys was implemented and stored in the Raspberry server's repository. The system's operational workflow follows these steps: the virtual sensors configured in AWS IoT Core connect and publish to the MQTT server (Raspberry), which periodically generates measurements of variables including temperature, humidity, and luminosity. The data consists of randomized values with variable parameters. The Raspberry device, functioning as an intermediate node, receives these messages via MQTT and immediately appends a transmission timestamp, precisely recording when the data was processed. After establishing the connection with the sensors, the script running on the Raspberry Pi generates randomized simulated measurements of temperature, humidity, and light. These data points are then written directly into the MySQL databases on each server, accompanied by their respective timestamps, ensuring a detailed and accurate reading history. The Fog server, physically closer to the Raspberry Pi, receives the data first, enabling faster processing and reducing latency associated with transmission. Meanwhile, the Cloud server—located remotely—also receives the measurements but with inherently longer transmission times due to geographical distance and network characteristics.

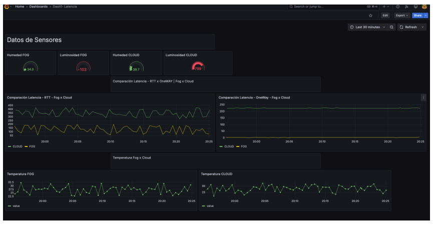
5 Results:

Fog Computing latency was significantly lower than Cloud latency. We tested two models (RTT and ONEWAY). Fog processing occurs closer to the devices, which significantly reduces transmission and storage latency. Results show the average time between a packet's departure from a sensor and its return plus subsequent database registration in the Fog environment ranged between 57 ms and 140 ms using RTT measurement. In contrast, the Cloud environment showed an average latency between 300 ms and 400 ms - considerably higher than Fog Computing using the same methodology. In OneWay packet transmission tests, the average time between departure and database registration for Fog processing ranged between 3-4 ms, while Cloud processing ranged between 220-225 ms.

Result Panel:
To present the collected sample data and carry out the necessary comparisons, we installed Grafana, which has a graphical interface with time stamps and temporal graphs. Grafana uses a graphical interface on a WEB server and connects to databases from both the cloud server and the fog server.

5 and 30 Minutes Menasurement.





6 Related Work:

In [3] The author presents cloud computing as an extension of cloud computing, highlighting its distinctive characteristics such as low latency, on the other hand [4] demonstrates the importance of taking the process to the limit in critical services. In [5] the author makes a comparison between clouds and clouds and use a tool to measure latency. In almost all studies, the topic of latency is of utmost importance. Its importance is clear in environments with a shortage of Internet links or, on the other hand, in environments that require a quick or immediate response. Reducing latency is crucial to success. By bringing processing to the edge of the network, processing it and then sending it to the cloud, we significantly help reduce network usage.

7 Conclusion:

In this article, we seek to present the results of results of experimental tests demonstrating that the use of Fog Computing significantly reduces latency in comparison with an approach based exclusively on Cloud Computing, thereby validating our initial hypothesis. This difference is calculated using key metrics, such as round-trip time (RTT) and response time with methodology (ONEWAY), which confirms that local processing in the cloud cover minimizes delays in transmission and data storage.

References

[1] E. A. Q. Montoya, S. F. J. Colorado, W. Y. C. Muñoz, and G. E. C. Golondrino, "Propuesta de una arquitectura para agricultura de precisión soportada en IoT," *Revista Ibérica de Sistemas e Tecnologias de Informação*, no. 24, pp. 39–56, 2017.

[2] A. V Dastjerdi and R. Buyya, "Fog Computing: Helping the Internet of Things Realize Its Potential," *Computer (Long Beach Calif)*, vol. 49, no. 8, pp. 112–116, 2016, doi: 10.1109/MC.2016.245.

[3] M. A. Orozco, "Análisis de factibilidad en la implementación de un sistema de control para el monitoreo del proceso de fermentación del vino (Doctoral dissertation, SIPI).," 2020. Accessed: May 31, 2023. [Online]. Available: https://repositorio.cetys.mx/bitstream/60000/1102/1/Orozco%20Armando_Proyecto%20final.pdf

[4] C. V Raghavendran, A. Patil, G. N. Satish, M. Shanmukhi, and B. Madhuravani, "Challenges and opportunities in extending cloud with fog computing," *Int J Eng Technol*, vol. 7, no. 439, pp. 142–146, 2018.

[5] R. Mahmud, K. Ramamohanarao, and R. Buyya, "Latency-aware application module management for fog computing environments," *ACM Transactions on Internet Technology (TOIT)*, vol. 19, no. 1, pp. 1–21, 2018.