ISBN: 978-628-95207-8-1. ISSN: 2414-6390. Digital Object Identifier: 10.18687/LACCEI2024.1.1.924

Quake100: A Neural Network-Based Application for Predicting Earthquakes in Peru

Carlos Quinto Huamán, PhD[1], Paulo Arce Oré, Student[1], Saul Arones Hernández, Student[1], Luis Paredes Silvestre [1], Sebastián Inga Rodríguez[1], and Gladys Madeleine Rojas Cangahuala, PhD[2]

[1]Universidad Privada del Norte, Lima, Perú, carlos.quinto@upn.pe, n00248456@upn.pe, n00236156@upn.pe, n00246231@upn.pe, n00302409@upn.pe

[2] Grupo de Investigación en Ciberseguridad, IoT e Inteligencia Artificial (GriCIA), Instituto Científico y Tecnológico del Ejército, Lima, Perú, grojasc@icte.edu.pe

***Abstract*– Currently, seismic events are recurrent in Peru, causing human losses and instilling fear among the population. This is primarily due to the absence of an early warning system or prediction platform that could foresee such tectonic events. Given Peru's geological location in a tectonically active area, earthquakes pose a significant threat. The main objective of this work is to implement a mobile application based on neural networks to predict the occurrence of earthquakes within a 100-day interval. The aim is to provide relevant information for risk management and disaster preparedness. This study involved the collection and preparation of a comprehensive set of historical seismic data, incorporating features such as magnitude, depth, location, and chronological sequence of events. Preliminary results indicate that neural networks have promising potential to generate reliable predictions of seismic events in Peru.**

In summary, this proposal contributes to the intersection of seismology and neural networks by suggesting a method for predicting seismic events in Peru using neural networks. Despite the remaining challenges, this study offers a promising path towards strengthening early warning systems and reducing seismic risk in the region. Continuing to integrate real-time data and improving neural network models can have a significant impact on the safety and resilience of Peruvian communities against seismic events in the future.

Keywords-- Neural Networks, Seismic Risks, Historical Data, Forecast, Seismic Movements.

I. Introduction

In Peru, the Geophysical Institute of Peru (IGP) is dedicated to observing and recording the physical, chemical, geological, and petrological manifestations of active volcanoes. The institute conducts scientific research with the aim of contributing to disaster prevention. According to IGP data, Peru is globally recognized as one of the nations with the highest seismic risk due to its location in the Pacific Ring of Fire, an area where over 85% of the Earth's accumulated energy is released through mantle convection processes. In this context, a report [1] asserts that the country is ill-prepared to face a high-magnitude earthquake, mainly due to the prevalence of informal economy contributing to unregulated constructions. This includes individuals constructing houses on a minimal budget without hiring specialists for geotechnical, structural, and architectural plans.

The interest in earthquake prediction is widespread, especially in regions with high seismic activity. Despite technological advances and increased understanding of Earth's behavior, precise earthquake prediction remains challenging. Studying seismic patterns and measuring changes in the Earth's crust deformation are two techniques currently used for earthquake prediction. However, these methods have limitations and cannot guarantee precise and reliable earthquake predictions. Consequently, using machine learning techniques, the development of a neural network capable of forecasting the probability of an earthquake in a specific geographic region becomes a valuable tool to mitigate earthquake consequences.

The main objective of this proposal is to increase the accuracy of seismic event predictions in Peru, including magnitude and location. This will be achieved by leveraging data collected by the IGP to construct an effective prediction model. The aim is also to significantly reduce the destructive impact of seismic disasters in the nation while simultaneously preparing the population for these natural events.

This work is structured into six sections, with the introduction being the first. Section II details related works found at the national and international levels. Section III presents the proposal for an application to forecast seismic events. Section IV provides results and discussion. Finally, Section V presents the conclusions.

II. Overview of Earthquake Forecasting Using Neural Networks

Understanding the intricate dynamics of seismic activity is crucial for implementing effective earthquake risk mitigation strategies [2]. Machine learning techniques, particularly neural networks, have proven to be powerful tools for accurately assessing seismic hazard [3]. Neural networks in seismology marked a significant turning point, heralding the convergence of artificial intelligence and geophysical research. In the past, the seismic domain, characterized by complex data patterns, posed analytical challenges that often exceeded the capabilities of conventional algorithms. It was in this context that neural networks emerged as a promising source of potential [4].

A neural network is based on a machine learning mechanism inspired by the human brain, allowing it to establish non-linear relationships between input and output variables. One of the main advantages of this machine learning technique is its ability to analyze, classify, and process information in complex patterns. The connection between different nodes within a neural network is established through synapses, and each node is assigned a different weight depending on its impact on the desired prediction [5]. Data in an input layer or node remains unchanged, meaning it is not subjected to operations; it is simply distributed throughout the network so that subsequent layers can make use of it and obtain the desired result. Hidden layers act as intermediaries between input and output layers; they receive data from the previous layer, process it, and transmit it to the next layer until reaching the output layer. The output layer typically consists of a single neuron, holding the final result [6].

Normalization of input data involves applying a mathematical operation to ensure that the input data falls within a range of 0 to 1 or -1 to 1. If this technique is not applied, the input will have an additional effect on the neuron. The mathematical equation for normalizing input data is as follows:

$$y = \frac{(x - x_{min})(d2 - d1)}{(x_{max}) - (x_{min})} + d1 \quad ===>\text{Equation 1}$$

Donde:
x = Value to be normalized
(x_{max}), (x_{min}) = Range in which the variable x is located
(d_1, d_2) = Range to which the data of variable x will be scaled

The activation function is responsible for obtaining the output value of a neuron based on a certain threshold, determining whether it should or should not transmit the obtained result to the next neuron [7]. The formula used is as follows:

$$\sum (w.x) + b \qquad ===>\text{Equation 2}$$

Where:
w = Weight assigned to each node
x = Input value of each node
b = Bias, a number that encourages some neurons to activate more easily than others.

Despite the existence of many activation functions, there are three that are most commonly used: (i) Threshold Function: This function returns a value of zero as long as the weighted sum is less than the specified threshold; otherwise, the function returns a value of 1. (ii) Sigmoid Function: It is typically used in networks with multiple layers or networks with continuous signals. (iii) Hyperbolic Tangent Function: Similar to the previous function, it is also used in networks with continuous signals, and its main characteristic is that it can return negative values, as the input values are transformed to a scale of [-1, 1]. It is very useful in recurrent neural networks.

In this context, there are several research studies that have attempted to address the prediction of seismic events. In [8], the exploration of the application of Deep Neural Networks (DNNs) in seismic data analysis was proposed, highlighting their ability to provide a higher level of abstraction and, consequently, improve model generalization. In [9], significant success was achieved in seismic discrimination, surpassing 99% accuracy, through the use of Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN). This involved distinguishing between quarry blasts and tectonic sources using event catalogs and sensor spectrograms. In [10], the effectiveness of Long Short-Term Memory (LSTM) networks in predicting the intensity function of temporal Epidemic-Type Aftershock Sequences (ETAS) was demonstrated. This emphasizes the adaptability of LSTM networks for precise temporal predictions in seismic activity. In [11], a research study was conducted with the aim of predicting the likelihood of seismic events and their characteristics using Machine Learning techniques. The author claims in the mentioned work that the techniques applied in the project are efficient in predicting upcoming events based on historical data. The use of machine learning techniques is deemed highly useful for solving various predictive problems. In [12], a probabilistic neural network application is proposed to predict the magnitude of major earthquakes. The architecture consists of an input layer, two hidden layers, and an output layer defined as a competition layer. Seismicity indicators were represented in each node of the input layer, and a Gaussian function was applied to each node in the first hidden layer. The dataset used was from the Southern California Earthquake Data Center, divided into seven groups based on Richter scale magnitudes. Results were achieved for predicting earthquakes with magnitudes between 4.5 and 6, while facing challenges in forecasting earthquakes above 6. In [13], the authors evaluated the predictive capacity of earthquake magnitudes in the Hindukush region using sensitivity and specificity metrics. They employed various machine learning techniques with a dataset spanning from April 1977 to December 2013, extracted from the Earthquake Studies Center and the United States Geological Survey. The techniques included a pattern recognition neural network and a Recurrent Neural Network (RNN). The authors concluded that the results were satisfactory, surpassing levels achieved in previous research.

In Peru, research related to earthquake prediction using machine learning techniques has been conducted. An example of this is the work carried out by the Geophysical Institute of Peru (IGP), where a neural network model was developed for the real-time detection and classification of earthquakes. The results obtained showed a success rate of 99.18%, with better performance compared to the classical STA/LTA algorithm, exhibiting 20% fewer false positives out of 1000 [14]. In [15], the author detected behavioral patterns in regions with seismic concentration through the analysis of time series using data from the years 2017 and 2018 in Peruvian territory. In this initial exploration, Cluster Analysis was applied to identify groups or geographic areas with proximity in seismic occurrence. A relationship was observed between the magnitude of earthquakes over time, evaluated based on nearby geographical zones, and the magnitude of the previous earthquake. Consequently, the ARIMA (1,1,0) model was applied, which incorporates lag and difference to eliminate trends, excluding the presence of a moving average. Eight geographical regions with concentrated seismic activity were identified. Among the notable findings, it was highlighted that in the Arequipa-Tacna and Lima-Ica areas, earthquake magnitudes in relation to the occurrence time align with the ARIMA (1,1,0) model. Additionally, it was confirmed that in the Arequipa-Tacna region, the depth of earthquakes in relation to the occurrence time also fits the ARIMA (1,1,0) model. This analysis was conducted using information provided by the Geophysical Institute of Peru, including details about time, latitude, altitude, magnitude, depth, among others, accessible on the institution's website.

In general, previous works in Peru and abroad have demonstrated the utility of machine learning techniques in earthquake prediction and risk assessment, laying the foundation for further research in this field.

Previous research in Peru and abroad has shown the usefulness of machine learning techniques in earthquake prediction and risk assessment. This has set the stage for the development of new investigations in this field.

III. DEVELOPMENT OF THE PROPOSAL

Due to Peru's geological location in a tectonically active area, earthquakes pose a significant and ongoing threat. In this regard, to address the need for improving the accuracy of seismic event prediction, Quake100 is introduced. It is a mobile application based on neural networks designed to forecast earthquakes in Peru. Specifically, it harnesses the power of neural networks to predict the occurrence of earthquakes over a forecasting period of 100 days.

The development of this project has been divided into two phases: (i) Model construction; and (ii) implementation of the mobile application. The Algorithm 1 presents the two main processes of the proposal. Figure 1 illustrates the stages of the proposal's development.

Algorithm 1: Develop_Project

Input: Project Development Plan

Result: Application for Predicting Earthquakes

1. **Procedure** Develop_Project
2.   **Function** Model Construction
3.   Develop the model
4.   **Function** Mobile Application Implementation
5.   Develop the application
6. **End procedure**

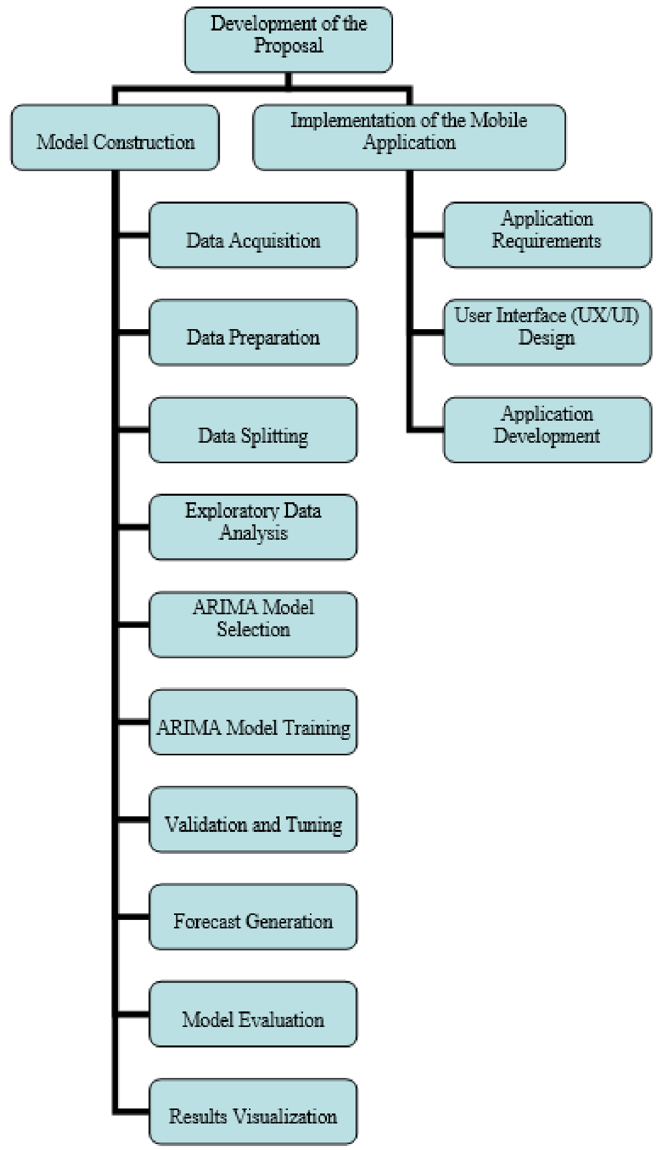

Fig. 1 Development Stages of the Proposal

A. *Model Construction*

In this stage, the power of Python and the ARIMA (Auto Regressive Integrated Moving Average) method is harnessed to construct predictive models for seismic events. This statistical technique is ideal for time series modeling and forecasting, making it particularly suitable for this project.

The model construction unfolds through the following steps:

- Data Acquisition: Historical data on seismic events in Peru, including information on magnitude, depth, location, and occurrence date, is gathered. This data is stored in a format suitable for analysis in Python. The dataset is available at https://www.igp.gob.pe/servicios/aceldat-peru/sismos-historicos.

- Data Preparation: Seismic data undergoes cleaning and preprocessing using Python. This involves handling outliers, addressing missing data, and ensuring that the data can be effectively utilized in ARIMA models.
- Data Splitting: The data is divided into training sets (80%) and testing sets (20%). The training set is employed to fit the ARIMA model, while the test set is reserved for evaluating the model's performance.
- Exploratory Data Analysis: In Python, exploratory analyses are conducted to gain a deeper understanding of the characteristics of seismic data. This includes time series visualizations, histograms, and autocorrelation analysis.
- ARIMA Model Selection: Utilizing autocorrelation and partial autocorrelation analysis (ACF and PACF), the optimal order of the ARIMA model (p, d, q) is determined. This process involves identifying the autoregressive (AR) component, the differencing (I) component, and the moving average (MA) component.
- ARIMA Model Training: The ARIMA model is trained on the training set using the statsmodels library. Parameters are estimated, and the model's adequacy is checked.
- Validation and Adjustment: The ARIMA model undergoes validation on the test set to assess its performance. Parameters are adjusted if necessary to enhance forecast accuracy.

The Algorithm 2 presents the Earthquake Prediction Pipeline based on historical seismic data from Peru. It begins by preprocessing the data to handle outliers and missing values. Then, it constructs two models: an ARIMA model and a neural network model. For the ARIMA model, the algorithm performs data splitting to create training and testing sets, followed by exploratory data analysis to understand the seismic data characteristics. Autocorrelation and partial autocorrelation analyses are conducted to select the optimal ARIMA model parameters. The chosen model is trained on the training set and validated on the testing set, with parameters adjusted if necessary. In parallel, the neural network model is constructed. Data splitting is performed again for training, and the neural network architecture is defined, including an input layer, two hidden layers with 64 and 32 neurons utilizing ReLU activation, and an output layer for earthquake magnitude prediction. Grid search with cross-validation is employed to find the best hyperparameters, iterating through different sets of hyperparameters and evaluating performance on validation folds. The neural network is then trained on the entire training set using the best hyperparameters. Subsequently, both models are validated on the testing set, and their performances are compared using evaluation metrics such as mean squared error (MSE) and mean absolute error (MAE). Finally, the model with the best performance for earthquake prediction is selected, and the algorithm returns the chosen model.

Algorithm 2: Earthquake Prediction Pipeline

Input: *L*: Historical earthquake data for Peru (data)

Result: Best selected model *(L)*

Procedure Earthquake Prediction Pipeline (*L*)
  Function Arima Model Construction ( )
    Split data into training and testing sets.
    Analyze data characteristics.
    Select ARIMA model using ACF and PACF analysis.
    Train ARIMA model.
    Validate and adjust ARIMA model.
  Function Neural Network Model Construction
    Split data for neural network training.
    Define neural network architecture.
    Perform Grid Search with Cross Validation
      Initialize best_score.
      For each hyperparameter set:
        Split training data into K folds.
        Train neural network on folds.
        Evaluate on validation fold.
        Calculate average performance metric.
        Update best hyperparameters if performance improves.
      Train neural network on full training set with best hyperparameters.
  Validate neural network model on testing set.
  Function Model Evaluation ( )
    Compare ARIMA and neural network performance.
    Select best performing model
  Return selected model
End procedure

Figure 2 illustrates the neural network configuration during this stage. The model creation is depicted with 2 hidden layers containing 64 and 32 neurons, respectively, along with an output layer. This configuration was chosen for its optimal forecasting efficiency. Metrics used include Mean Squared Error (MSE) and Mean Absolute Error (MAE).

```
model_ann = Sequential()
model_ann.add(Dense(64, activation='relu', input_shape=(X.shape[1],)))
model_ann.add(Dense(32, activation='relu'))
model_ann.add(Dense(1))
model_ann.compile(optimizer='adam', loss='mse', metrics=['mae'])
```

Fig. 2 Creation of the Neural Network

Mean Squared Error is a metric used to assess the quality of a model's predictions. It is calculated by taking the difference between each model prediction and the corresponding real value, squaring these differences, and then averaging these squared values. The equation is expressed as follows:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad ===>\text{Equation 3}$$

Where:

n =es el número de observaciones en el conjunto de prueba.

yi= represents the real value of the variable.

y^i= represents the model's prediction for that observation.

Mean Absolute Error (MAE) is another metric for evaluating the accuracy of a model's predictions. Unlike MSE, MAE takes the absolute value of the differences between the model's predictions and the real values and then calculates the average of these absolute values. The equation has the same variables as MSE:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \quad ===>\text{Equation 3}$$

- Generation of Forecast: Using the adjusted ARIMA model, forecasts for future seismic events are generated. These forecasts can provide valuable information for risk management. For visualization of the results on a map of the Peruvian territory, the Basemap library from Matplotlib is utilized. Subsequently, latitude and longitude coordinates are converted into X coordinates. A figure is created, and the location points are plotted on the map.
- Model Evaluation: Evaluation metrics (MSE and MAE) are employed to measure the accuracy of the ARIMA model forecasts. Table 1 presents the evaluation results for the neural network (NN) model and the ARIMA model. For the neural network model, the best hyperparameters consist of a two-layer architecture with 64 and 32 neurons, respectively. The evaluation metric used to select this model was the Mean Squared Error (MSE), with a value of 0.0089. As for the ARIMA model, the best hyperparameters are not explicitly defined, as they depend on the specific values of p, d, and q that are optimal for each dataset. Similar to the neural network model, the evaluation metric used to select the ARIMA model was the Mean Squared Error (MSE), with a value of 0.090.

TABLE I
TECHNOLOGIES USED FOR TESTING

RESULTS	VALUES
Best Hyperparameters (NN)	(64, 32)
Best Metric (NN)	MSE
Best Metric Value (NN)	0.0089
Best Hyperparameters (ARIMA)	(p, d, q)
Best Metric (ARIMA)	MSE
Best Metric Value (ARIMA)	0.0090

- Results Visualization: The results are generated to represent the forecasts in comparison with the actual data, facilitating result interpretation. For this purpose, two external libraries, reverse_geocoder and pycountry, are utilized to obtain the city and country based on the latitude and longitude of each record in a DataFrame. A function is defined that takes a row parameter and uses the latitude and longitude information to look up the city and country using the reverse_geocoder library. Then, the function uses the pycountry library to convert the two-letter country code into the full name of the country. The function returns a text string containing the name of the city and the country.

B. Implementation of the Mobile Application

At this stage, the implementation includes the UI Design, UX Design, and obtaining prediction results phases. For the implementation of the mobile application, the following phases were carried out:

B.1 Application Requirements

In this step, functional and non-functional requirements of the mobile application were defined. This includes deciding what features the application should have, how it should interact with users, and how it will integrate with other components of the project. Below are the main requirements of the application:

- REQ-1: Efficient Response Time: The system must generate predictions in a reasonable time for quick responses.

- REQ-2: Accuracy and Confidence in Predictions: Train the neural network with representative data to ensure reliable predictions.
- REQ-3: Attractive Graphical Interface: Design a visually appealing and easy-to-understand user interface.
- REQ-4: Data Security and Privacy: Protect seismic data and personal information during transfer and storage.
- REQ-5: Scalability and Load Handling: Ensure that the system handles large volumes of data and simultaneous users without performance loss.
- REQ-6: Clear and Accessible Documentation: Provide detailed documentation on the operation, installation, and configuration for non-expert users.
- REQ-7: Simplified Maintenance and Updates: Maintain modular and well-commented code to facilitate future updates and maintenance.
- REQ-8: Compatibility with Python Libraries: Use widely accepted Python libraries to ensure project continuity and maintainability.

B.2 User Interface Design (UX/UI)

A design for an intuitive and easy-to-use user interface is proposed. This involves creating screen designs, icons, and navigation flows that allow users to quickly access the necessary information and functions.

B.3 Application Development

The Android development environment (Android Studio) and the Kotlin programming language were used to create the mobile application. During this phase, all planned features were implemented, including seismic data visualization, event notifications, and interaction with the virtual assistant.

IV. RESULTS AND DISCUSSION

A. Results

The dataset instances have allowed us to observe all recorded seismic events from 1996 to the present. In Figure 3, a map displaying the geographical distribution of seismic events is presented. For the construction of the graph, the map projection is established using the Basemap class, which takes the coordinates of the lower-left corner and upper-right corner of the map, as well as the latitude of the baseline and resolution as arguments. Next, latitude and longitude coordinates are extracted from the data and converted to the map projection using the (longitudes, latitudes) method of Basemap.

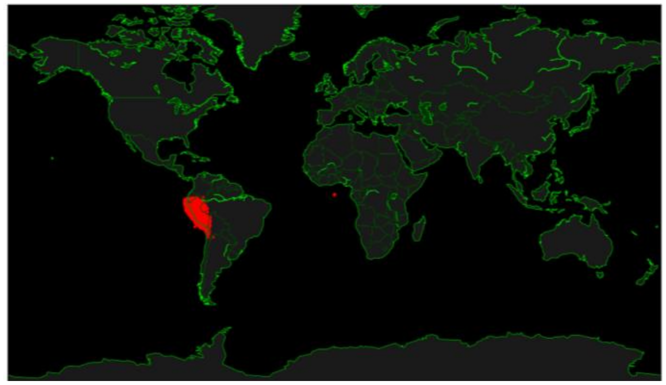

Fig. 3 Geographic Map of Seismic Events from 1960 to Present

In Figure 4, the forecasts obtained by the neural network are observed, starting from August 19, 2023, detailing the magnitude, latitude, longitude, and the predicted city.

Depth	Date	Magnitude	Latitude	Longitude	Ciudad
28414.1	2023-08-19	4.4	-8.99	-79.81	Salaverry, Peru
29316.1	2023-08-20	4.7	-16.61	-72.94	El Cardo, Peru
29351.6	2023-08-21	4.0	-5.81	-81.28	Sechura, Peru
29359.9	2023-08-22	4.4	-6.56	-76.29	Shapaja, Peru
29347.2	2023-08-23	4.2	-16.21	-73.23	Urasqui, Peru
29362.4	2023-08-24	4.2	-15.49	-72.47	Viraco, Peru
29355.7	2023-08-26	4.0	-16.11	-71.67	Yura, Peru
29357.8	2023-08-27	4.3	-10.43	-74.54	Bajo Pichanaqui, Peru
29354.7	2023-08-28	4.1	-17.69	-70.49	Sama Grande, Peru
29351.6	2023-08-29	4.6	-14.80	-76.20	Ocucaje, Peru
29361.0	2023-08-30	4.2	-15.81	-73.32	Iquipi, Peru
29348.5	2023-08-31	0.0	0.00	0.00	Takoradi, Ghana
29349.3	2023-09-01	4.9	-13.46	-74.69	Totos, Peru
29353.7	2023-09-02	4.0	-17.77	-70.11	Palca, Peru
29348.9	2023-09-03	4.0	-11.33	-79.03	Paramonga, Peru
29366.6	2023-09-04	4.5	-8.26	-80.03	Santiago de Cao, Peru

Fig. 4 Obtained Forecasts

Upon evaluating the neural network model, the Mean Squared Error (MSE) was calculated, reaching a value of 0.0091. A low MSE value indicates that the model's predictions are very close to the actual values, suggesting precision in terms of earthquake magnitude.

The achieved Mean Absolute Error (MAE) value was 0.0678. A low MAE value indicates that the model's predictions have small absolute errors compared to the actual values. In other words, the model tends to make predictions close to the real values in terms of earthquake magnitude. In this context, the low MSE and MAE values in the project suggest that the prediction model is effective and accurate in estimating seismic events.

Figure 5 displays the forecasts made by the neural network for the next 100 days. It is observed that the generated forecasts are located within the territory of Peru or very close to it. This outcome confirms that the neural network model, in conjunction with the ARIMA model, can learn the patterns and generate outputs consistent with the used data.

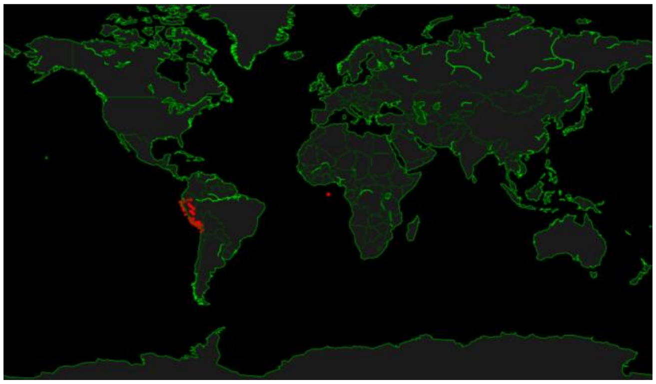

Fig. 5 Map of the next 100 forecasted seismic events

In Figure 6, real values and forecasted values can be observed. The range is organized by decades, so the last value on the X-axis is 2020. The orange line represents the predicted values, following the pattern of events recorded in the dataset used. That is, there is no noticeable difference between the predicted values and the events recorded in the IGP dataset. The plot function from matplotlib.pyplot is used to create the graph.

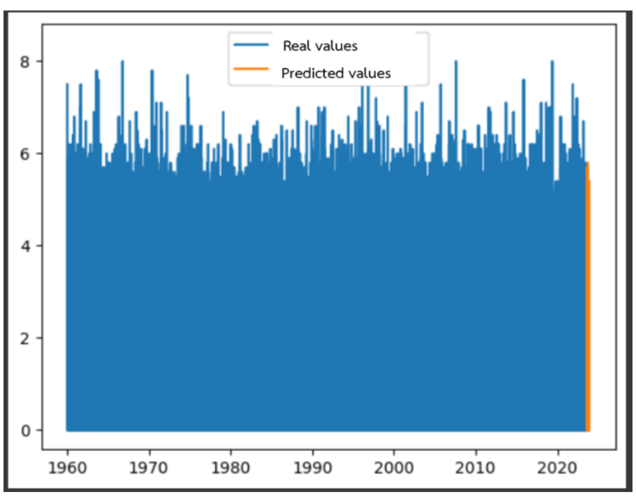

Fig. 6 Plot of Real and Predicted Values

To assess the model's effectiveness, a sample of predictions from Figure 4 was taken, and the event record provided by IGP was downloaded to create a comparative table. Figure 7 shows the comparison between real and predicted data.

Forecast				Events Recorded by IGP			
Date	Magnitude	Latitude	Longitude	Date	Magnitude	Latitude	Longitude
19/08/2023	4.4	-8.99	-79.81	19/08/2023	4	-9	-79.43
20/08/2023	4.7	-16.61	-72.94	20/08/2023	4.2	-16.68	-71.87
21/08/2023	4	-5.81	-81.28	21/08/2023	4	-3.09	-80.19
22/08/2023	4.4	-6.56	-76.29	22/08/2023	4	-9.48	-79.12
23/08/2023	4.2	-16.21	-73.23	23/08/2023	5.2	-20.73	-69.54
24/08/2023	4.2	-15.49	-72.47				
25/08/2023	0	0	0				
26/08/2023	4	-16.11	-71.67	26/08/2023	4.6	-15.64	-69.6
27/08/2023	4.3	-10.43	-74.54	27/08/2023	4	-18.59	-71
28/08/2023	4.1	-17.69	-70.49	28/08/2023	4.3	-16.41	-72.62
29/08/2023	4.6	-14.8	-76.2				
30/08/2023	4.2	-15.81	-73.32	30/08/2023	4.2	-4.41	-77.93
31/08/2023	0	0	0	31/08/2023	5.1	-1.49	-78
1/09/2023	4.9	-13.46	-74.69				
2/09/2023	4	-17.77	-70.11	2/09/2023	4.8	-15.66	-75.05
3/09/2023	4	-11.33	-79.03				
4/09/2023	4.5	-8.26	-80.03				
5/09/2023	4.5	-15.78	-71.76	5/09/2023	4	-9.21	-79.47
6/09/2023	5.7	-7.99	-74.56				
7/09/2023	5.6	-20.59	-70.6				
8/09/2023	0	0	0	8/09/2023	4.5	-7.96	-74.61
9/09/2023	4.4	-5.66	-77.07	9/09/2023	4.7	-16	-74.73
10/09/2023	4.6	-15.96	-75.33	10/09/2023	4.1	-8.39	-76.38
11/09/2023	5.2	-15.78	-71.72	11/09/2023	4.6	-15.7	-74.81
12/09/2023	4.3	-15.76	-71.73				
13/09/2023	4.8	-15.8	-71.74	13/09/2023	4	-14.35	-71.45
14/09/2023	4	-15.8	-71.88				
15/09/2023	5.4	-16.32	-74.19	15/09/2023	4	-17.35	-70.87
16/09/2023	4	-15.92	-72.93				
17/09/2023	4.1	-8.38	-80.26				
18/09/2023	4.1	-4.43	-79.5	18/09/2023	4	-4.93	-80.83
19/09/2023	4	-8.3	-80.15	19/09/2023	4	-15.38	-74.53
20/09/2023	4	-14.36	-75.77	20/09/2023	4	-8.77	-79.82
21/09/2023	4.5	-16.27	-74.19	21/09/2023	4	-5.68	-76.71
22/09/2023	0	0	0	22/09/2023	4.3	-12.72	-76.13
23/09/2023	4.5	-13.56	-74.8	23/09/2023	4.1	-18.71	-70.15
24/09/2023	0	0	0	24/09/2023	4	-16.45	-72.51
25/09/2023	4.3	-4.2	-81.91	25/09/2023	4.2	-16.32	-74.53
26/09/2023	4.4	-15.49	-75.39	26/09/2023	4	-15.2	-75.08
27/09/2023	5.1	-15.79	-74.77	27/09/2023	4	-10	-78.95
28/09/2023	4.5	-13.09	-76.78	28/09/2023	4.8	-15.8	-74.34

Fig. 7 Comparison between Prediction and Real Data

The forecasted values include the location (latitude and longitude) and the magnitude of the seismic event. On the left side are the forecasted values, and on the right side are the real values. In some cases, we see that the neural network correctly predicts the magnitude of the earthquake and comes quite close to the location of the event; in other cases, we observe the opposite, where it has a significant degree of accuracy in predicting the location and comes quite close to the magnitude of the earthquake. Of course, the degree of accuracy is not absolute, and there are days when the predictions generate a result, yet no events are recorded for that specific day. It should be taken into consideration that the model is configured to generate forecasts for the next 100 days from the last record in the dataset used in the test. Therefore, it is a definition as such to generate forecasts for each day and not a prediction error.

The forecast results can be observed from any mobile device with the built application installed. The application can obtain the forecasted data and display it to the user through a simple forecast request. Initially, the user will need to register with a personal email account. Figure 8 shows the login window of the QUAKE100 application.

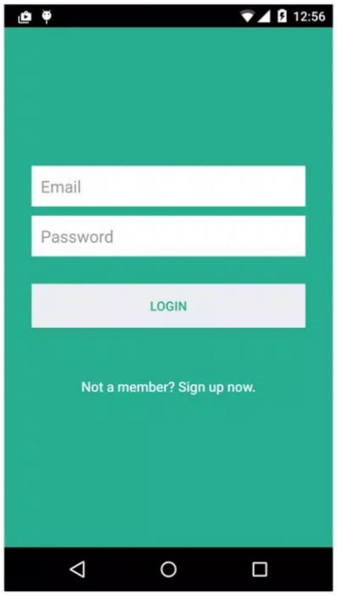

Fig. 8 Application Login

Once the user has been identified, the main window of QUAKE100 will be displayed, welcoming the user and then initiating the forecast query. Figure 9 shows the main screen of QUAKE100.

Finally, the user will be able to observe the earthquake forecast for the next 100 days in an easy and intuitive way. (See Figure 10).

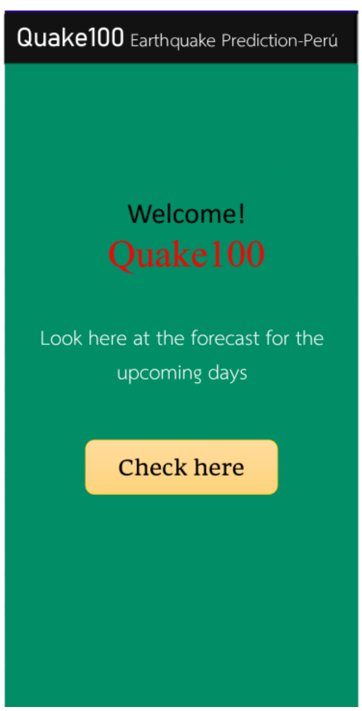

Fig. 9 Main screen of QUAKE100

Fig. 10 Results Presentation Screen

B. Discussion

The obtained results from the neural network model and its integration with the ARIMA model indicate promising capabilities in earthquake prediction. The low values of Mean Squared Error (MSE) and Mean Absolute Error (MAE) suggest that the model's predictions closely align with actual seismic events, demonstrating precision in magnitude estimation. Additionally, the geographical distribution of forecasted seismic events within the Peruvian territory, as shown in Figure 5, further supports the model's ability to capture patterns and generate location-specific predictions.

The comparative analysis in Figure 7 highlights instances where the neural network accurately predicts either the magnitude or the location of seismic events. The variability in accuracy can be attributed to the complex nature of seismic activity and the multitude of factors influencing earthquake occurrence. Nevertheless, the overall performance, as reflected in the evaluation metrics, signifies the model's effectiveness in providing valuable forecasts.

The user interface and functionality of the QUAKE100 application offer an accessible platform for users to interact with earthquake forecasts. The login process, as depicted in Figure 8, ensures personalized access to forecast data.

The forecast presentation screen provides an intuitive interface for users to initiate and view earthquake predictions for the next 100 days (Figure 10). While the developed model demonstrates promising results, it is essential to acknowledge the inherent uncertainties associated with earthquake prediction. Earthquakes are complex phenomena influenced by various factors, and despite advancements in machine learning and data analysis, complete accuracy in predicting seismic events remains a challenging task. Continuous refinement of the model through the incorporation of real-time data and ongoing validation will contribute to its reliability.

Moreover, the success of the QUAKE100 application relies on the integration of accurate and up-to-date data. Regular updates to the dataset, coupled with improvements in data quality and quantity, will enhance the model's predictive capabilities. Collaboration with geological and seismological institutions for data sharing and validation can further strengthen the application's reliability.

.

VI. CONCLUSIONS

This study highlights the significant potential of predictive models in assessing and forecasting seismic events, especially in tectonically active regions like Peru. Although earthquake prediction remains a challenge due to the chaotic nature of these events, the results presented here demonstrate encouraging advancements in this field.

The combination of modeling techniques, such as the use of neural networks and the ARIMA method, has proven promising in generating acceptable forecasts for future seismic events. The ability of these techniques to identify patterns in complex datasets and generate accurate predictions is highlighted as a significant step towards improving earthquake prediction accuracy.

The results obtained through the implementation of the Quake100 project show a low margin of error in both the training and testing phases, supporting the effectiveness of the developed models. Error indicators, such as Mean Squared Error (MSE) and Mean Absolute Error (MAE), show low values, suggesting that the models are capable of generating predictions close to the actual values of seismic events.
Additionally, visualizing the results on geographical maps provides a clear and accessible representation of the generated forecasts, facilitating data interpretation and their application in seismic risk management.

While this study presents promising advancements, it is important to consider limitations and areas for improvement. Model accuracy can still be enhanced by incorporating real-time data and continuously validating with historical records and real observations. Furthermore, collaboration with geological and seismological institutions can enrich datasets and improve forecast reliability.

In summary, this study highlights the potential of predictive models in seismic event prediction and underscores the importance of ongoing research in this field to enhance accuracy and responsiveness in seismic risk management in vulnerable regions like Peru.

ACKNOWLEDGMENT

The authors would like to express their gratitude to Universidad Privada del Norte and Instituto Científico y Tecnológico del Ejército for enabling the development and dissemination of the proposal.

REFERENCES

[1] UDEP. (2021, July). ¿Estamos preparados para afrontar un sismo en el Perú? [Online]. Available: https://www.udep.edu.pe/hoy/2021/07/estamos-preparados-para-afrontar-un-sismo-en-el-peru/
[2] Bottari C., Capizzi P., & Sortino F. Unraveling the Seismic Source in Archaeoseismology: A Combined Approach on Local Site Effects and Geochemical Data Integration. Heritage. 2024; 7(1):427-447. https://doi.org/10.3390/heritage7010021
[3] Gitis, V. (2019). Machine Learning Methods for Seismic Hazards Forecast. Geosciences. 2019; 9(7):308. https://doi.org/10.3390/geosciences9070308
[4] Falcone, R., et al. (2022). Artificial neural network for technical feasibility prediction of seismic retrofitting in existing RC structures,
[5] Structures. Structures, 1220-1234. https://doi.org/10.1016/j.istruc.2022.05.008.
[6] Villada, F., Muñoz, N., & Garcia-Quintero, E. (2016). Redes Neuronales Artificiales aplicadas a la Predicción del Precio del Oro. Inf. tecnol., 27(5). DOI: http://dx.doi.org/10.4067/S0718-07642016000500016
[7] Fontalvo, T. J., & De la Hoz, E. J. (2018). Diseño e Implementación de un Sistema de Gestión de la Calidad ISO 9001:2015 en una Universidad Colombiana. Form. Univ., 11(1). DOI: http://dx.doi.org/10.4067/S0718-50062018000100035
[8] Calvo, D. (2018). Función de Activación en Redes Neuronales. [Online]. Available: https://www.diegocalvo.es/funcion-de-activacion-redes-neuronales/
[9] Kislov, K. V., & Gravirov. Deep artificial neural networks as a tool for the analysis of seismic data. Seism. Instrum., 2018.
[10] Linville, et al. (2019). Achieving superior seismic discrimination through Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN). Spatial Stat.
[11] Plaza, et al. (2019). Temporal Epidemic-Type Aftershock Sequences (ETAS) intensity prediction in Chile using Long Short-Term Memory (LSTM) networks. Seismol. Res. Lett.
[12] Alba Vega, D. A., & Calle Jara, J. F. (2020). Repositorio Institucional de la Universidad de Guayaquil. [Online]. Available: http://repositorio.ug.edu.ec/handle/redug/48862
[13] Hojjat Adeli and Ashif Panakkat (2009). A probabilistic neural network for earthquake magnitude prediction. Journal Neural Network, 22(7), 1018-1024.
[14] Asim, K. M., Martinez Alvarez, F., Basit, A., & Iqbal, T. (2017). Predicting the magnitude of an earthquake in the Hindukush region using machine learning techniques. Nat Hazards, 85, 471–486.
[15] Sotelo, J. Z. (2021). Implementación de una Red Neuronal Convolucional para la clasificación de ruido sísmico y señales sísmicas. IEEE Congreso Estudiantil de Eléctronica y Electricidad (INGELECTRA), Valparaiso, Chile.
[16] Risco Franco, C. A. (2021). Análisis de series de tiempo de datos de sismos en el Perú 2017-2018. Revista IECOS, 19, 121–134. DOI: https://doi.org/10.21754/iecos.v19i0.1173