Research Article GU J Sci 37(3): 1172-1188 (2024) DOI:10.35378/gujs.1364529

Gazi University

Journal of Science

http://dergipark.gov.tr/gujs

Hybrid Deep Learning Model for Earthquake Time Prediction

Anıl UTKU[1, *], Muhammet Ali AKCAYOL[2]

[1]*Munzur University, Computer Engineering Department, 62100, Tunceli, Türkiye*
[2]*Gazi University, Computer Engineering Department, 06480, Ankara, Türkiye*

Highlights

- A comparative analysis for earthquake time prediction is presented.
- A hybrid model was developed using CNN and GRU models.
- The developed hybrid model was more successful than traditional methods.

Article Info

Received: 21 Sep 2023
Accepted: 27 Feb 2024

Keywords

Earthquake
Deep learning
Machine learning
CNN
GRU

Abstract

Earthquakes are one of the most dangerous natural disasters that have constantly threatened humanity in the last decade. Therefore, it is extremely important to take preventive measures against earthquakes. Time estimation in these dangerous events is becoming more specific, especially in order to minimize the damage caused by earthquakes. In this study, a hybrid deep learning model is proposed to predict the time of the next earthquake to potentially occur. The developed CNN+GRU model was compared with RF, ARIMA, CNN and GRU. These models were tested using an earthquake dataset. Experimental results show that the CNN+GRU model performs better than others according to MSE, RMSE, MAE and MAPE metrics. This study highlights the importance of predicting earthquakes, providing a way to help take more effective precautions against earthquakes and potentially minimize loss of life and material damage. This study should be considered an important step in the methods used to predict future earthquakes and supports efforts to reduce earthquake risks.

1. INTRODUCTION

Today, many processes and events that affect people's lives in different fields must be predicted before they happen. Risk analysis makes it possible to determine whether events will occur at a particular time and to take or prevent protective measures against them [1]. However, there are natural disasters such as tsunamis, hurricanes, floods, volcanic eruptions, and earthquakes that even today's technology cannot prevent. People cannot stop such natural disasters but minimize deaths and economic losses with precautions and rapid interventions [2].

Natural disasters are often noted to be unpredictable and non-analyzable. Earthquakes are considered the most dangerous and destructive natural disasters that may lead to other natural disasters, such as tsunamis, avalanches, and landslides [3]. Aside from this, earthquakes may usually occur suddenly and cause people to be caught off guard. For this reason, it is one of the most devastating natural disasters that causes severe injuries and deaths. Additionally, they also cause significant economic losses by damaging buildings and infrastructure. Predicting earthquakes is essential in terms of public safety, but it is a very challenging issue in terms of seismology.

For all these reasons, earthquake forecasting is a critical issue for safety [4]. Researchers have tried to discover the leading earthquakes and the abnormal events that preceded these disasters. Many possible signs were studied, including leading earthquakes, electromagnetic anomalies, changes in groundwater levels, and unusual animal behavior [2].

*Corresponding author, e-mail: anilutku@munzur.edu.tr

Today, there still needs to be a consensus among earthquake researchers on whether earthquake prediction is possible. However, the successful application of machine learning methods in many different application fields can help predict earthquakes accurately before they happen [5].

Moustra et al. have presented an application of artificial neural networks for predicting seismic activities in Greece [6]. Two different studies were conducted in the study: the first is related to predicting the earthquake's magnitude, while the second is predicting the magnitude of the upcoming earthquake. The earthquake's magnitude prediction accuracy is 80.55%, and the accuracy of the seismic events is 58.02%.

Florido et al. have also analysed seismic time series data for Chile's four most active regions to find patterns between major earthquakes [7]. It is concluded that the proposed method, which has an average success rate of 70%, is suitable for other regions with similar seismicity.

Asencio-Cortes et al. applied artificial neural networks to predict the earthquake's magnitude in Tokyo [8]. Experimental studies were carried out for earthquakes larger than 5.0 in five datasets. The developed model was compared with C4.5 decision tree, k-Nearest Neighbors (kNN), Support Vector Machine (SVM), and Naive Bayes (NB) algorithms.

A Long-Short Time Memory (LSTM)-based model was developed by Wang et al. to learn the space-time relations between earthquakes in different places [1]. This LSTM model, with a two-dimensional input, was created to extract spatial-temporal relations between historical earthquake data. In simulations, earthquake data greater than 2.5 was used. The proposed model has significantly increased the system performance.

Kavianpour et al. proposed a new model using CNN and Bi-LSTM models to predict the maximum magnitude and number of earthquakes [9]. The proposed model was tested using a dataset obtained from different regions in China. Experimental results showed that the proposed model was more successful than DT, CNN, LSTM, MLP, RF and SVM.

Sadhukhan et al. presented a comparative analysis of LSTM, Bi-LSTM and transformer models for predicting the magnitude of the next possible earthquake [10]. In the study, abnormal temperature data and earthquake parameters selected according to the Richter scale were used to determine the seismic status of the regions. In the study, seismic data from Indonesia, Hindi Kush and Japan were used. Experimental results showed that LSTM was more successful than compared models.

Berhich et al. developed an attention-based LSTM model to predict earthquake magnitude, time, and location [11]. Japan seismic data between 1900 and 2021 were used as the dataset. The dataset was divided into geographical clusters exhibiting similar earthquake characteristics using the K-Means algorithm. The developed model was compared with LSTM and ANN. Experimental results showed that the MSE value of an attention-based LSTM model was approximately 60% lower than other models.

To contribute with the recent trends in addressing the predictive potential of earthquakes, this study aims to develop and investigate a hybrid deep learning model for predicting the time of the next earthquake. The developed hybrid CNN+GRU model was compared with AutoRegressive Integrated Moving Average (ARIMA), Random Forest (RF), CNN, and GRU.

2. MATERIAL METHOD

The aim of this study is to use a series of analytical and computational techniques to develop a more accurate and reliable method for time estimation of earthquakes. It is known that earthquakes pose serious threats worldwide and cause serious damage to human life, infrastructure and economy. Therefore, taking timely warning and precautions against earthquakes is of critical importance. In this section, the data sources used, analysis methods and the configuration of the developed CNN+GRU model will be explained in detail. It will also be discussed how this work compares with other traditional and deep learning models and how its

performance is measured. This will provide a basic framework for understanding the basic methodology and results of the study.

2.1. Dataset

In this study, the earthquake dataset provided by the National Earthquake Information Center (NEIC) was used. This dataset contains records of each earthquake whose size is reported as 5.5 or higher from January 2, 1965, to December 30, 2016. There are 23.232 earthquake records in the dataset. The dataset used is publicly available via https://www.kaggle.com/usgs/earthquake-database?select=database.csv.

Table 1 shows the first five rows of the dataset.

Table 1. *The first five rows of the dataset*

Date	Time	Latitude	Longitude	Type	Depth
01/02/1965	13:44:18	19.246	145.616	Earthquake	131.6
01/04/1965	11:29:49	1.863	127.352	Earthquake	80
01/05/1965	18:05:58	-20.579	-173.972	Earthquake	20
01/08/1965	18:49:43	-59.076	-23.557	Earthquake	15
01/09/1965	13:32:50	11.938	126.427	Earthquake	15

The earthquake dataset consists of 23.232 rows and 10 columns.

2.2. Data Pre-processing

This section describes the steps for processing and preparing the earthquake dataset used. Data preprocessing is a critical step to obtain accurate and reliable results. Each feature in the dataset was carefully examined and missing or abnormal values were detected. Data on earthquakes comes in the form of time series data. Therefore, the dataset is made suitable for time series analysis. By using the sliding window method to organize and analyze time series data, time series data was transformed into a supervised learning problem. In the sliding window method, the previous time steps can be configured as input variables and the next time step can be configured as output variables. Figure 1 shows the sliding window method.

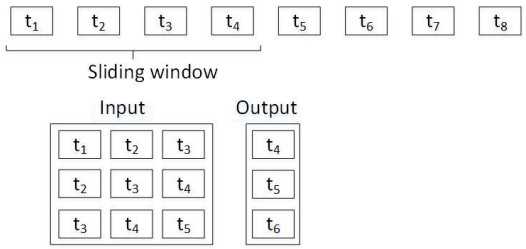

Figure 1. *Converting time series data into supervised learning problem*

As shown in Figure 1, the data at t_1, t_2 and t_3 are configured as input and the data at t_4 as the output. In this study, the window size is defined as 3. Then, the elapsed times between the earthquake events are calculated in seconds. The features to be used for earthquake time estimation are derived from the dataset. These features were chosen to better capture earthquake activity and increase the accuracy of the model. Additionally, features were created taking into account seasonal and trend components.

For the analyzes carried out on the dataset, normalization and standardization of the data were carried out. This ensured that different features were at the same scale and helped the model perform better. MinMaxSclaer was used for data normalization.

The dataset was divided into 67% training and 33% testing. 10% of the training data was used for validation. Validation data was used to optimize hyperparameters. As shown in Figure 2, parameter optimization was

performed using walk-forward validation. In this way, the model parameters with the lowest Mean Squared Error (MSE) value were determined.

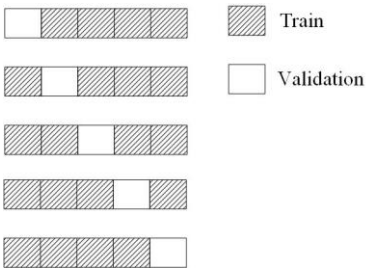

Figure 2. *Model validation*

Walk-forward validation is an adapted version of the cross-validation technique for time series. In walk-forward validation, the data in the validation set is processed and added to the test data. This process continues until all data in the validation set are processed and the parameters with the lowest MSE value are determined.

2.3. Baseline Models

RF is an ensemble method consisting of decision trees. RF is created by combining multiple decision trees [12]. These decision trees are trained on different subset samples of the dataset. Each tree attempts to capture different aspects of the dataset. RF performs random sampling and feature selection during training of each tree [13]. This allows each tree to be trained with different data and features, also reducing overlearning. RF makes stronger and more stable predictions by combining the predictions of many decision trees. This is quite effective for handling the complexity of time series data. RF provides a feature importance ranking that measures the contribution of each feature to the predictions [14]. This is useful for understanding which features are more predictive in time series data. RF is a powerful method that can be used for time series analysis and is often preferred to increase the accuracy of forecasts and address complexity in the dataset.

ARIMA is a statistical model widely used to analyze and forecast time series data. The AutoRegressive (AR) component of the ARIMA model refers to an autoregressive model based on data from past time steps. This means predicting future values using predictions from previous time steps of time series data [15]. The AR component is used to capture autocorrelation between data [16]. The I component of ARIMA addresses the non-stationary nature of the time series. The integration step makes the data series stationary by differencing or applying degrees of difference [17]. This eliminates the trend and seasonal components of the data series. The Moving Average (MA) component of the ARIMA model refers to a moving average model based on the values of the error terms in past time steps [18]. This component is used to capture random fluctuations and error terms.

ARIMA is effective for analyzing time series data and predicting future values, especially in datasets that contain trends, seasonality, and autocorrelation [19]. Determining the model requires the correct selection of parameters called (*p*, *d*, *q*). *p* denotes the number of autoregressive terms, *d* denotes the degree of integration, and *q* denotes the number of moving average terms. ARIMA is a powerful tool used to predict future values by taking into account the statistical properties of time series data [18]. With the ARIMA model, predictions are calculated using Equation (1)

$$y_t = \mu + \varphi_1 y_{t-1} + ... + \varphi_p y_{t-p} - \theta_1 e_{t-1} - ... - \theta_q e_{t-q} \ . \tag{1}$$

Here, y_t represents the predicted value, e_t represents the error at time *t*, φ_i and θ_j represent the coefficients. μ represents white noise, which is the number of AR(p) and MA(q) polynomials.

CNN, initially designed for image recognition tasks, have also proven to be effective for various applications, including time series analysis [20]. CNN process input data through convolutional layers. These layers slide a filter matrix over the data to detect different features, allowing the identification of temporal patterns in time series data. Pooling layers reduce the dimension of the data while preserving important information. This is crucial for summarizing the features of time series data and maintaining relevant details. At the end of the CNN, fully connected layers are used to classify or make predictions based on the learned features [21]. These layers combine the extracted features to analyze time series data and make forecasts. CNN can capture higher-level features by adding multiple convolutional layers. This is particularly useful for detecting complex patterns in time series data. CNN can be a powerful tool for addressing complexity in time series data and capturing temporal patterns [22]. However, designing a CNN model for time series data and tuning hyperparameters should be done carefully, taking into account the specific application and dataset.

CNN consists of a convolution layer, a pool layer and fully bound layers. The convolution layer allows the weight to be shared, while the lower sampling is used to reduce the size. For example, two-dimensional x, image first $x = \{x_1, x_2, ..., x_N\}$ is resolved into the form of a sequential input. To share weight, the convolution layer is defined as follows:

$$y_j = f\left(\sum_i K_{ij} \otimes x_i + b_j\right) . \tag{2}$$

Here, y_i is j. output for the convolution layer. K_{ij} refers to the convolutional nucleus with input map x_i . $\otimes$ refers to the discrete convolution operator and b_j refers to the bias variable. f indicates a typically hyperbolic tangent function.

GRU is equipped with gating mechanisms that help control the flow of information through the network [23]. These mechanisms consist of update and reset gates, which determine which information to retain and which to discard [24]. This helps mitigate the vanishing gradient problem commonly associated with traditional RNNs. GRU is particularly well-suited for capturing short-term dependencies and patterns in time series data [25]. GRU is computationally efficient compared to other RNN variants. This makes them suitable for training on large datasets and deploying in real-time applications. GRU can be used for various time series analysis tasks, including forecasting, anomaly detection, and sequence generation [26]. GRU has 2 inputs, current input (x_t) and previous hidden state (h_{t-1}). GRU produces vectors with values 0 and 1 with sigmoid activation functions using Equations (3) and (4) for updating

$$r_u = \sigma(W_u[h_{t-1}, x_t] + b_u) \tag{3}$$

$$r_u = \sigma(W_r[h_{t-1}, x_t] + b_r) . \tag{4}$$

Using Equations (3) and (4), it is determined which information from the previous hidden state (h_{t-1}) is required. Using Equation (5), a hidden state candidate (h'_{t-1}) is generated to identify stored information from the past

$$h'_{t-1} = \tanh(W_h[\Gamma_r * h_{t-1}, x_t] + b_h) . \tag{5}$$

Using Equation (6), a new hidden state (h_t) is generated, which depends on the update gate (Γ_u) and the hidden candidate state (h'_t)

$$h'_t = ((1 - \Gamma_u) * h_{t-1}) + (\Gamma_u * h'_t) . \tag{6}$$

y is calculated using the current hidden state (h_t) using Equation (7)

$$y_t = g(W_y h_t + b_y)\,. \tag{7}$$

2.4. Developed Hybrid Deep Learning Model

In this study, a hybrid deep learning model developed to obtain more precise and reliable results in earthquake time prediction. The model was developed by combining CNN and GRU architectures. This hybrid model stands out with its ability to effectively capture both the structural features of the original data and the patterns of time series data over time. The architecture of the developed model is seen in Figure 3.

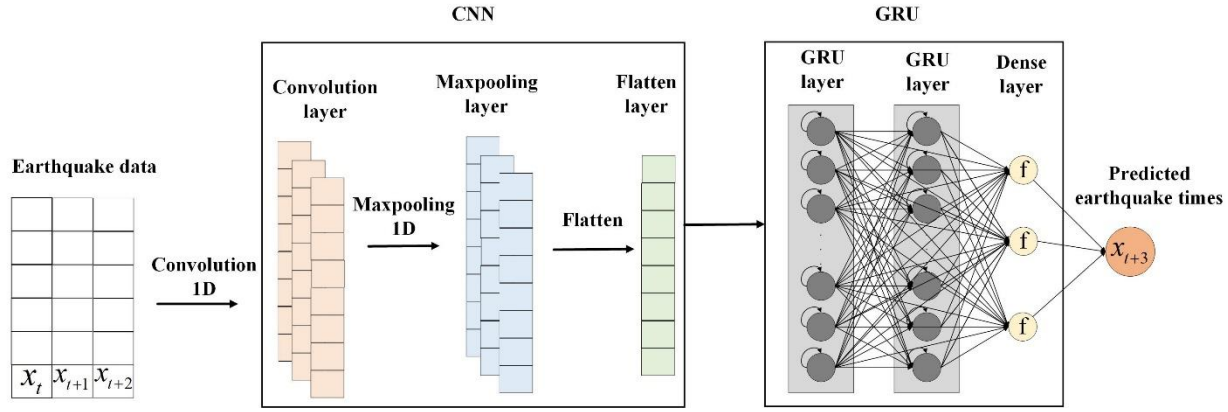

Figure 3. *The architecture of the developed model*

The model first starts with CNN layers to process the input data. CNN is used to capture and learn structural features of earthquake data over time. It may contain multiple convolutional and pooling layers. Convolutional layers extract visual features by transforming data into feature maps. Pooling layers perform dimensionality reduction and feature extraction.

The output of the CNN layers is passed to the GRU layers. GRU is used to capture short-term dependencies and patterns in time series data. GRU uses a unique gating mechanism to preserve and forget information from past time steps. This helps the model capture dynamics over time.

The final outputs from the GRU layers are passed to one or more fully connected (dense) layers. These layers combine features and compress the results for use in predictions. The outputs from the last fully connected layer are transmitted to the output layer, which is used to make earthquake time estimation. This layer produces the probability distribution of predictions.

While CNN is used to learn the structure of earthquake data, GRU provides an ideal structure to capture dependencies and patterns in past time steps. The performance of the model was evaluated especially with MSE, RMSE (Root Mean Squared Error), MAPE (Mean Absolute Percentage Error), and MAE (Mean Absolute Error) metrics, and better results were obtained compared to other traditional models. This model architecture aims to make more precise and reliable predictions by combining both structural features and the importance of time series data in earthquake time prediction.

The developed CNN+GRU hybrid model represents an important step in the field of earthquake time prediction. The model successfully integrates complex features to process original data and predict future earthquake times. Additionally, this model performs better compared to other traditional methods, making it possible to obtain more powerful and accurate results in earthquake time prediction. The developed model provides more effective preparation against future earthquakes by emphasizing the importance of predicting

earthquakes and taking precautions accordingly. This supports efforts to reduce earthquake risks and improves the safety of communities.

3. THE EXPERIMENTAL RESULTS

The main focus of this study is to develop an effective method for earthquake time prediction and compare this method with existing prediction models. Earthquakes are unpredictable natural disasters that can cause serious damage to human life and infrastructure. Therefore, developing an accurate and reliable model for earthquake time prediction is of great importance.

This section presents the experimental results of the traditional CNN and GRU models along with the RF and ARIMA models used as current prediction models. These basic models are methods used in previous studies in earthquake time estimation and are widely accepted in the literature. This study will present a framework using these baseline models to determine the performance of these baseline models and compare them with the proposed CNN+GRU hybrid model.

This section will also present preliminary information on why the proposed CNN+GRU model may perform better and highlight the potential superiority of this new approach in earthquake time prediction. Additionally, this comparative study will contribute to future earthquake time prediction studies by providing more information and a guiding approach to earthquake risk reduction efforts.

RF, ARIMA, CNN, GRU, and CNN+GRU were tested and compared. For each model, hyper-parameters were optimized using GridSearch. The MSE, RMSE, MAE, and MAPE were used to calculate the error value. The tests were run 10 times for each model, then average error values were calculated.

The earthquake dataset consists of 23232 events. The training dataset consists of 15565 events, and the test dataset consists of 7.667 events. The first 5 lines of the dataset created according to the elapsed time in seconds are shown in Table 2.

Table 2. *Elapsed times between earthquake events*

Event	Elapsed time
1	2745
2	1836
3	4363
4	1123
5	1443

The experimental results based on MSE are presented in Table 3 and Figure 4.

Table 3. *The experimental results according to MSE*

Test	RF	ARIMA	CNN	GRU	CNN+GRU
1	1610590.28	1558449.38	1473729.68	1418450.10	**1212268.12**
2	1610590.28	1558449.38	1473837.17	1421194.99	**1213058.62**
3	1610590.28	1558449.38	1474148.86	1422081.89	**1213575.12**
4	1610590.28	1558449.38	1474258.97	1422101.48	**1217778.78**
5	1610590.28	1558449.38	1474275.08	1422452.61	**1218862.21**
6	1610590.28	1558449.38	1474379.54	1422477.23	**1221256.32**
7	1610590.28	1558449.38	1474384.42	1422795.24	**1225789.70**
8	1610590.28	1558449.38	1474570.26	1422821.93	**1226366.28**
9	1610590.28	1558449.38	1474606.38	1423573.04	**1227327.13**
10	1610590.28	1558449.38	1474888.89	1424636.84	**1227375.18**
Average	1610590.28	1558449.38	1474307.92	1422258.53	**1220365.74**

The dataset used is a time series dataset. In time series, data is ordered chronologically according to a time

index. Table 3 shows the results obtained after running each model 10 times. Since ARIMA is a statistical model, RF had the same results in each running step because the decision trees would have the same statistical results on the same data. However, deep learning models have different results at each running step due to their complex structures and learnable parameters.

Table 3 and Figure 4 show that the best MSE of CNN+GRU is 1212268.12, the worst MSE is 1227375.18, and the average MSE is 1220365.74. For RF and ARIMA, the MSE values are 1610590.28 and 1558449.38, respectively. The average MSE for CNN and GRU was 1474307.92 and 1422258.53, respectively.

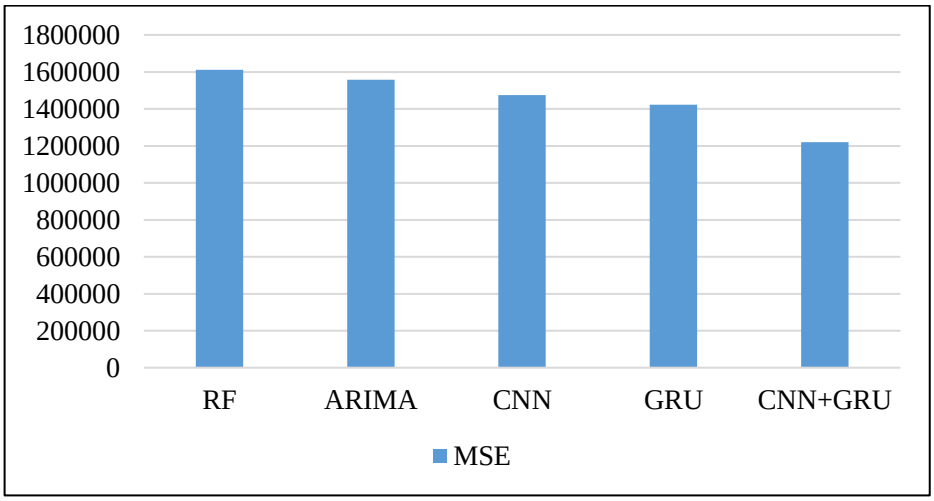

Figure 4. *The experimental results according to MSE*

Equations (8)-(11) were used to determine the Improvement of the CNN+GRU according to RF, ARIMA, CNN and GRU

$$\frac{\text{RF-(CNN+GRU)}}{\text{RF}}\times 100 \tag{8}$$

$$\frac{\text{ARIMA-(CNN+GRU)}}{\text{ARIMA}}\times 100 \tag{9}$$

$$\frac{\text{CNN-(CNN+GRU)}}{\text{CNN}}\times 100 \tag{10}$$

$$\frac{\text{GRU-(CNN+GRU)}}{\text{GRU}}\times 100 \cdot \tag{11}$$

The improvement of CNN+GRU according to MSE are shown in Table 4.

Table 4. *Improvement of CNN+GRU according to MSE metric*

	The worst value of CNN+GRU	The best value of CNN+GRU	Average value of CNN+GRU
Improvement according to RF (%)	23.79	24.73	24.22
Improvement according to ARIMA (%)	21.24	22.21	21.69
Improvement according to the worst value of CNN (%)	16.78	17.80	17.25
Improvement according to the best value of CNN (%)	16.71	17.74	17.19
Improvement according to the average value of CNN (%)	16.75	17.77	17.22

Improvement according to the worst value of GRU (%)	13.84	14.90	14.33
Improvement according to the best value of GRU (%)	13.47	14.53	13.96
Improvement according to the average value of GRU (%)	13.73	14.76	14.19

According to MSE values obtained from 10 tests, CNN+GRU was compared to RF, ARIMA, CNN, and GRU, and the results are shown in Table 4. The average improvement rate of the CNN+GRU in comparison to RF is 24.22%. The CNN+GRU has a 17.22% improvement rate when compared to CNN. In comparison to GRU, CNN+GRU has a 14.19% improvement rate.

The improvement rates of the average MSE of CNN+GRU relative to RF, ARIMA, CNN, and GRU are shown in Figure 5.

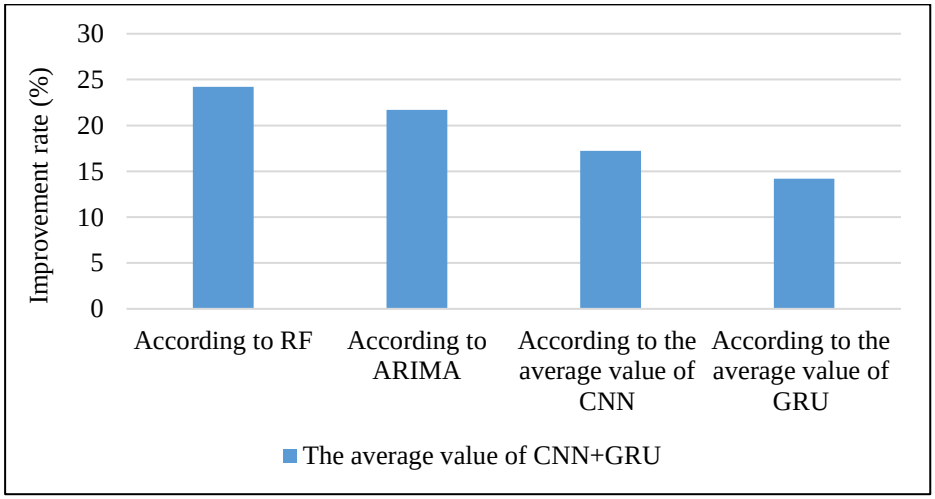

Figure 5. *Improvement of CNN+GRU according to MSE metric*

The experimental results based on RMSE are shown in Table 5 and Figure 6.

Table 5. *The experimental results according to RMSE*

Test	RF	ARIMA	CNN	GRU	CNN+GRU
1	1269.09	1248.37	1213.97	1190.98	**1101.03**
2	1269.09	1248.37	1214.02	1192.13	**1101.38**
3	1269.09	1248.37	1214.15	1192.51	**1101.62**
4	1269.09	1248.37	1214.19	1192.51	**1103.53**
5	1269.09	1248.37	1214.20	1192.66	**1104.02**
6	1269.09	1248.37	1214.24	1192.67	**1105.10**
7	1269.09	1248.37	1214.24	1192.80	**1107.15**
8	1269.09	1248.37	1214.32	1192.82	**1107.41**
9	1269.09	1248.37	1214.33	1193.13	**1107.84**
10	1269.09	1248.37	1214.45	1193.58	**1107.86**
Average	1269.09	1248.37	1214.21	1192.57	**1104.69**

Table 5 shows that the best RMSE of CNN+GRU is 1101.03, the worst RMSE is 1107.86, and the average RMSE is 1104.69. For RF and ARIMA, the RMSE values are 1269.09 and 1248.37, respectively. The average RMSE values of CNN and GRU are 1214.21 and 1192.57, respectively.

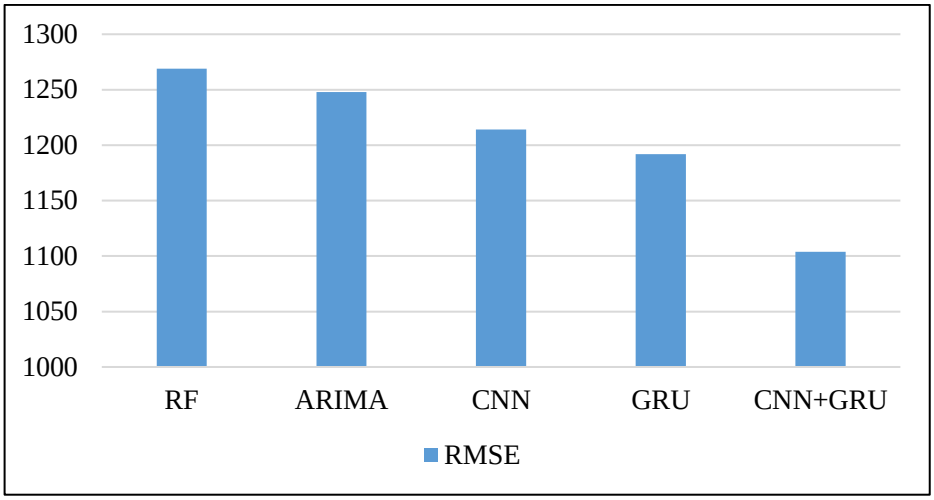

Figure 6. The experimental results according to RMSE

Equations (8)-(11) were used to calculate the improvement rate of the CNN+GRU in comparison to RF, ARIMA, CNN, and GRU. The improvement of CNN+GRU according to the RMSE metric is shown in Table 6.

Table 6. *Improvement of CNN+GRU according to RMSE metric*

	The worst value of CNN+GRU	The best value of CNN+GRU	Average value of CNN+GRU
Improvement according to RF (%)	12.70	13.24	12.95
Improvement according to ARIMA (%)	11.25	11.80	11.50
Improvement according to the worst value of CNN (%)	8.77	9.33	9.03
Improvement according to the best value of CNN (%)	8.74	9.30	9.00
Improvement according to the average value of CNN (%)	8.76	9.32	9.01
Improvement according to the worst value of GRU (%)	7.18	7.75	7.44
Improvement according to the best value of GRU (%)	6.97	7.55	7.24
Improvement according to the average value of GRU (%)	7.11	7.67	7.36

According to RMSE values obtained from 10 tests, CNN+GRU was compared with RF, ARIMA, CNN, and GRU, as shown in Table 6. Compared to RF, the worst value of CNN+GRU has improved by 12.70%, the best value of CNN+GRU has improved by 13.24%, and the average value of CNN+GRU has improved by 12.95%. According to ARIMA, the worst value of CNN+GRU has improved by 11.25%, the best value of CNN+GRU has improved by 11.80%, and the average value of CNN+GRU has improved by 11.50%.

According to the worst value of CNN, the worst value of CNN+GRU has improved by 8.77%, the best value of CNN+GRU has improved by 9.33%, and the average value of CNN+GRU has improved by 9.03%. According to the best value of CNN, the worst value of CNN+GRU has improved by 8.74%, the best value of CNN+GRU has improved by 9.30%, and the average value of CNN+GRU has improved by 9.00%. According to the average value of CNN, the worst value of CNN+GRU has improved by 8.76%, the best value of CNN+GRU has improved by 9.32%, and the average value of CNN+GRU has improved by 9.01%.

According to the worst value of GRU, the worst value of CNN+GRU has improved by 7.18%, the best value of CNN+GRU has improved by 7.75%, and the average value of CNN+GRU has improved by 7.44%.

According to the best value of GRU, the worst value of CNN+GRU has improved by 6.97%, the best value of CNN+GRU has improved by 7.55%, and the average value of CNN+GRU has improved by 7.24%. According to the average value of GRU, the worst value of CNN+GRU has improved by 7.11%, the best value of CNN+GRU has improved by 7.67%, and the average value of CNN+GRU has improved by 7.36%. The improvement rates of the average RMSE value of CNN+GRU relative to the average values of RF, ARIMA, CNN, and GRU are shown in Figure 7.

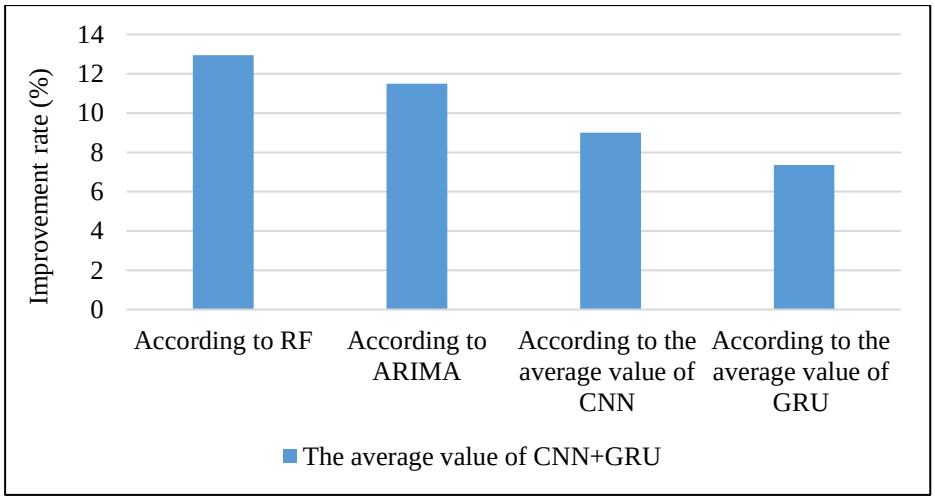

Figure 7. *Improvement of CNN+GRU according to RMSE metric*

The experimental results based on MAE are shown in Table 7 and Figure 8.

Table 7. *The experimental results according to MAE*

Test	RF	ARIMA	CNN	GRU	CNN+GRU
1	945.86	943.82	941.01	887.46	**804.42**
2	945.86	943.82	941.10	890.49	**804.48**
3	945.86	943.82	941.38	890.60	**804.84**
4	945.86	943.82	941.49	890.96	**812.90**
5	945.86	943.82	941.51	891.32	**815.02**
6	945.86	943.82	941.60	891.63	**816.36**
7	945.86	943.82	941.61	891.64	**817.00**
8	945.86	943.82	941.77	892.11	**817.34**
9	945.86	943.82	941.81	892.13	**817.83**
10	945.86	943.82	942.09	893.12	**817.90**
Average	945.86	943.82	941.53	891.14	**812.80**

Table 7 shows that the best MAE of CNN+GRU is 804.42, the worst MAE is 817.90, and the average MAE is 812.80. As RF and ARIMA are deterministic models, the MAE values are 945.86 and 943.82, respectively. The average values of MAE for CNN and GRU are 941.53 and 891.14, respectively.

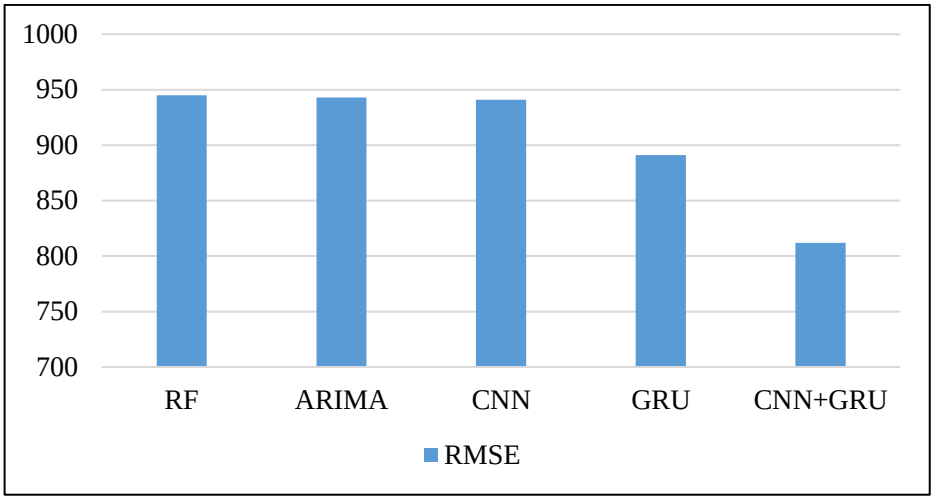

Figure 8. *The experimental results according to MAE*

Equations (8)-(11) were used to calculate the improvement rate of the CNN+GRU according to RF, ARIMA, CNN, and GRU. The improvement of CNN+GRU according to the MAE metric is shown in Table 8.

Table 8. *Improvement of CNN+GRU according to MAE metric*

	The worst value of CNN+GRU	The best value of CNN+GRU	Average value of CNN+GRU
Improvement according to RF (%)	13.52	14.95	14.06
Improvement according to ARIMA (%)	13.34	14.76	13.88
Improvement according to the worst value of CNN (%)	13.18	14.61	13.72
Improvement according to the best value of CNN (%)	13.08	14.51	13.62
Improvement according to the average value of CNN (%)	13.13	14.56	13.67
Improvement according to the worst value of GRU (%)	8.42	9.93	8.99
Improvement according to the best value of GRU (%)	7.83	9.35	8.41
Improvement according to the average value of GRU (%)	8.25	9.73	8.79

According to MAE values obtained from 10 tests, CNN+GRU was analyzed and compared with RF, ARIMA, CNN, and GRU. According to RF, the worst value of CNN+GRU has improved by 13.52%, the best value of CNN+GRU has improved by 14.95%, and the average value of CNN+GRU has improved by 14.06%. According to ARIMA, the worst value of CNN+GRU has improved by 13.34%, the best value of CNN+GRU has improved by 14.76%, and the average value of CNN+GRU has improved by 13.88%.

According to the worst value of CNN, the worst value of CNN+GRU has improved by 13.18%, the best value of CNN+GRU has improved by 14.61%, and the average value of CNN+GRU has improved by 13.72%. According to the best value of CNN, the worst value of CNN+GRU has improved by 13.08%, the best value of CNN+GRU has improved by 14.51%, and the average value of CNN+GRU has improved by 13.62%. According to the average value of CNN, the worst value of CNN+GRU has improved by 13.13%, the best value of CNN+GRU has improved by 14.56%, and the average value of CNN+GRU has improved by 13.67%.

According to the worst value of GRU, the worst value of CNN+GRU has improved by 8.42%, the best value of CNN+GRU has improved by 9.93%, and the average value of CNN+GRU has improved by 8.99%.

According to the best value of GRU, the worst value of CNN+GRU has improved by 7.83%, the best value of CNN+GRU has improved by 9.35%, and the average value of CNN+GRU has improved by 8.41%. According to the average value of GRU, the worst value of CNN+GRU has improved by 8.25%, the best value of CNN+GRU has improved by 9.73%, and the average value of CNN+GRU has improved by 8.79%.

The improvement rate of the MAE value of CNN+GRU relative to RF, ARIMA, CNN, and GRU is shown in Figure 9.

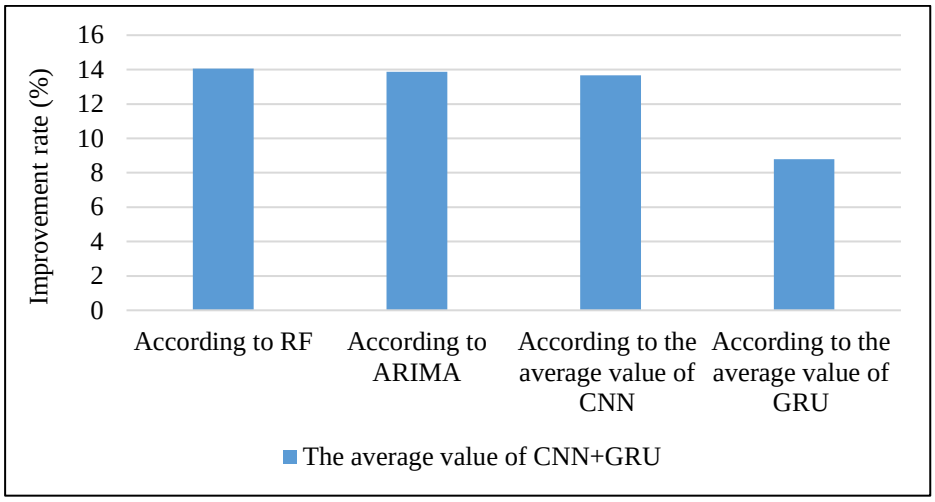

Figure 9. *Improvement of CNN+GRU according to MAE metric*

The experimental results based on MAPE are shown in Table 9 and Figure 10.

Table 9. *The experimental results according to MAPE*

Test	RF	ARIMA	CNN	GRU	CNN+GRU
1	1100.04	1097.66	1054.88	859.95	**779.13**
2	1100.04	1097.66	1064.42	867.14	**780.41**
3	1100.04	1097.66	1063.96	874.23	**780.96**
4	1100.04	1097.66	1077.21	858.36	**781.10**
5	1100.04	1097.66	1050.63	868.47	**781.36**
6	1100.04	1097.66	1051.78	866.59	**782.45**
7	1100.04	1097.66	1047.54	869.34	**782.55**
8	1100.04	1097.66	1062.23	867.11	**782.60**
9	1100.04	1097.66	1070.95	864.33	**785.04**
10	1100.04	1097.66	1064.39	865.12	**786.05**
Average	1100.04	1097.66	1060.79	866.06	**782.16**

Table 9 shows that the best MAPE of CNN+GRU is 779.13, the worst MAPE is 786.05, and the average MAPE is 782.16. As RF and ARIMA are deterministic models, the MAPE values are 1100.04 and 1097.66, respectively. The average MAPE values for CNN and GRU are 1060.79 and 866.06, respectively.

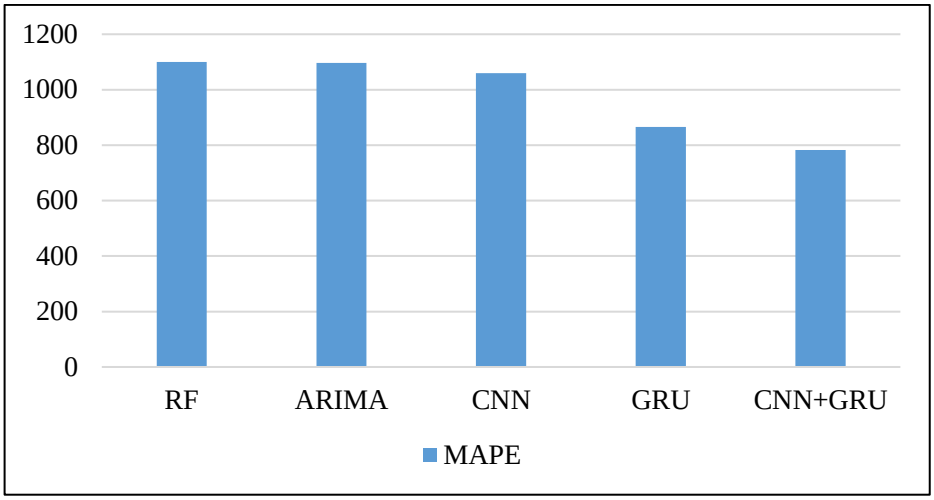

Figure 10. *The experimental results according to MAPE*

Equations (8)-(11) were used to calculate the improvement rate of the CNN+GRU according to RF, ARIMA, CNN, and GRU. The improvement of CNN+GRU according to the MAPE metric is shown in Table 10.

Table 10. *The improvement of CNN+GRU according to MAPE*

	The worst value of CNN+GRU	The best value of CNN+GRU	Average value of CNN+GRU
Improvement according to RF (%)	28.54	29.17	28.89
Improvement according to ARIMA (%)	28.38	29.01	28.74
Improvement according to the worst value of CNN (%)	27.02	27.67	27.39
Improvement according to the best value of CNN (%)	24.96	25.62	25.55
Improvement according to the average value of CNN (%)	25.89	26.55	26.26
Improvement according to the worst value of GRU (%)	10.08	10.87	10.53
Improvement according to the best value of GRU (%)	8.42	9.23	8.87
Improvement according to the average value of GRU (%)	9.23	10.03	9.68

According to MAPE values obtained from 10 tests, CNN+GRU was analyzed and compared with RF, ARIMA, CNN, and GRU. According to RF, the worst value of CNN+GRU has improved by 28.54%, the best value of CNN+GRU has improved by 29.17%, and the average value of CNN+GRU has improved by 28.89%. According to ARIMA, the worst value of CNN+GRU has improved by 28.38%, the best value of CNN+GRU has improved by 29.01%, and the average value of CNN+GRU has improved by 28.74%.

According to the worst value of CNN, the worst value of CNN+GRU has improved by 27.02%, the best value of CNN+GRU has improved by 27.67%, and the average value of CNN+GRU has improved by 27.39%. According to the best value of CNN, the worst value of CNN+GRU has improved by 24.96%, the best value of CNN+GRU has improved by 25.62%, and the average value of CNN+GRU has improved by 25.55%. According to the average value of CNN, the worst value of CNN+GRU has improved by 25.89%, the best value of CNN+GRU has improved by 26.55%, and the average value of CNN+GRU has improved by 26.26%.

According to the worst value of GRU, the worst value of CNN+GRU has improved by 10.08%, the best value of CNN+GRU has improved by 10.87%, and the average value of CNN+GRU has improved by

10.53%. According to the best value of GRU, the worst value of CNN+GRU has improved by 8.42%, the best value of CNN+GRU has improved by 9.23%, and the average value of CNN+GRU has improved by 8.87%. According to the average value of GRU, the worst value of CNN+GRU has improved by 9.23%, the best value of CNN+GRU has improved by 10.03%, and the average value of CNN+GRU has improved by 9.68%.

The improvement rate of the MAPE value of CNN+GRU relative to RF, ARIMA, CNN and GRU is shown in Figure 11.

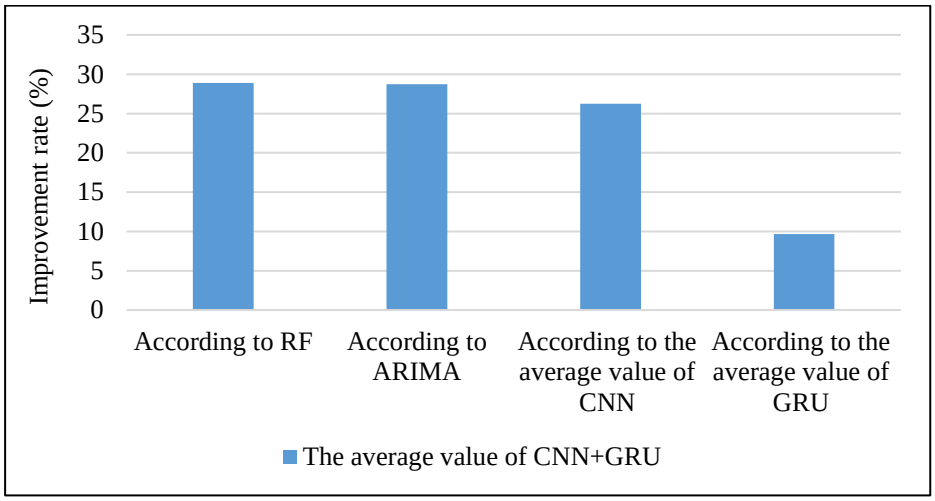

Figure 11. *Improvement of CNN+GRU according to MAPE metric*

As shown in Tables 4-10 and Figures 4-11, CNN+GRU is more successful than RF, ARIMA, CNN, and GRU in predicting the time of the next earthquake.

4. CONCLUSIONS

In this study, CNN+GRU hybrid model was developed to predict the next earthquake time. In the developed hybrid model, CNN and GRU models were combined. In CNN+GRU, CNN is used for feature extraction and GRU is used for prediction. CNN+GRU was compared with RF, ARIMA, CNN, and GRU models. The models were applied to an earthquake dataset.

The combination of the prominent and effective features of the GRU and CNN models has enabled the developed hybrid model to be successful. CNN is an effective model, especially in the feature extraction phase. GRU is an effective recurrent neural network model in the learning and prediction stages. The GRU has gates that regulate the flow of information. In this way, it ensures that long-term dependencies are remembered. The limitations of the developed model can be explained by the limitations of CNN and GRU. CNN can run slow for operations like max-pooling as the number of layers' increases. The GRU can be affected by gradient vanishing in cases where the dataset is extensive and the weights are not adjusted properly.

Experimental results showed that the CNN+GRU model was more successful than the compared models. Additionally, the results of CNN+GRU were found to improve the results of the compared models in terms of all performance metrics. Experimental studies using MSE, RMSE, MAE and MAPE metrics showed that CNN+GRU is more successful in predicting earthquake times. CNN+GRU provides flexibility to identify spatial patterns and capture changes in time series. In this way, it provides both spatial and temporal information that is important for earthquake time prediction. CNN+GRU reduces the training and prediction times of the model. It allows the model to be trained faster and prediction times shorter, especially when large datasets are used.

Experimental results show that CNN+GRU is more successful than traditional models. The results obtained will allow earthquake predictions to be made earlier and more accurately than traditional models. It is very important to develop early warning systems for vital natural disasters such as earthquakes. The effectiveness of the CNN+GRU model is promising for future research.

In future studies, using a comprehensive earthquake dataset, earthquakes' magnitude, occurrence times, and coordinates can be predicted with different parameters. In addition, the performance of the models can be evaluated by developing different hybrid deep-learning models.

CONFLICTS OF INTEREST

No conflict of interest was declared by the authors.

REFERENCES

[1] Wang, Q., Guo, Y., Yu, L., Li, P., "Earthquake prediction based on spatio-temporal data mining: An LSTM network approach", IEEE Transactions on Emerging Topics in Computing, 8(1): 148-158, (2017).

[2] Galkina, A., Grafeeva, N., "Machine learning methods for earthquake prediction: A survey", In Proceedings of the Fourth Conference on Software Engineering and Information Management (SEIM-2019), Saint Petersburg, Russia, (2019).

[3] Pribadi, K.S., Abduh, M., Wirahadikusumah, R.D., Hanifa, N.R., Irsyam, M., Kusumaningrum, P., Puri, E., "Learning from past earthquake disasters: The need for knowledge management system to enhance infrastructure resilience in Indonesia", International Journal of Disaster Risk Reduction, 64, (2021).

[4] Erzin, Y., Cetin, T., "The use of neural networks for the prediction of the critical factor of safety of an artificial slope subjected to earthquake forces", Scientia Iranica, 19(2): 188-194, (2012).

[5] Asim, K. M., Martínez-Álvarez, F., Basit, A., Iqbal, T., "Earthquake magnitude prediction in Hindukush region using machine learning techniques", Natural Hazards, 85(1): 471-486, (2017).

[6] Moustra, M., Avraamides, M., Christodoulou, C., "Artificial neural networks for earthquake prediction using time series magnitude data or seismic electric signals", Expert systems with applications, 38(12): 15032-15039, (2011).

[7] Florido, E., Martínez-Álvarez, F., Morales-Esteban, A., Reyes, J., Aznarte-Mellado, J. L., "Detecting precursory patterns to enhance earthquake prediction in Chile", Computers & Geosciences, 76: 112-120, (2015).

[8] Asencio-Cortés, G., Martínez-Álvarez, F., Troncoso, A., Morales-Esteban, A., "Medium–large earthquake magnitude prediction in Tokyo with artificial neural networks", Neural Computing and Applications, 28(5): 1043-1055, (2017).

[9] Kavianpour, P., Kavianpour, M., Jahani, E., and Ramezani, A., "A cnn-bilstm model with attention mechanism for earthquake prediction", The Journal of Supercomputing, 1-33, (2023).

[10] Sadhukhan, B., Chakraborty, S., Mukherjee, S., Samanta, R. K., "Climatic and seismic data-driven deep learning model for earthquake magnitude prediction", Frontiers in Earth Science, 11, 1082832, (2023).

[11] Berhich, A., Belouadha, F. Z., Kabbaj, M. I., "An attention-based LSTM network for large earthquake prediction", Soil Dynamics and Earthquake Engineering, 165: 107663, (2023).

[12] Katuwal, R., Suganthan, P. N., Zhang, L., "An ensemble of decision trees with random vector functional link networks for multi-class classification", Applied Soft Computing, 70: 1146-1153, (2018).

[13] Akhiat, Y., Manzali, Y., Chahhou, M., Zinedine, A., "A new noisy random forest based method for feature selection", Cybernetics and Information Technologies, 21(2): 10-28, (2021).

[14] Zhang, F., Yang, X., "Improving land cover classification in an urbanized coastal area by random forests: The role of variable selection", Remote Sensing of Environment, 251, (2020).

[15] Wang, Z., Lou, Y., "Hydrological time series forecast model based on wavelet de-noising and ARIMA-LSTM", 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chengdu, China, (2019).

[16] Khan, F. M., Gupta, R., "ARIMA and NAR based prediction model for time series analysis of COVID-19 cases in India", Journal of Safety Science and Resilience, 1(1): 12-18, (2020).

[17] Patil, P., "A comparative study of different time series forecasting methods for predicting traffic flow and congestion levels in urban networks", International Journal of Information and Cybersecurity, 6(1): 1-20, (2022).

[18] Schaffer, A. L., Dobbins, T. A., Pearson, S. A., "Interrupted time series analysis using autoregressive integrated moving average (ARIMA) models: a guide for evaluating large-scale health interventions", BMC medical research methodology, 21(1): 1-12, (2021).

[19] Dhyani, B., Kumar, M., Verma, P., Jain, A., "Stock market forecasting technique using arima model", International Journal of Recent Technology and Engineering, 8(6): 2694-2697, (2020).

[20] Ren, H., Xu, B., Wang, Y., Yi, C., Huang, C., Kou, X., Zhang, Q., "Time-series anomaly detection service at Microsoft", Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, USA, (2019).

[21] He, Q., Zheng, Y. J., Zhang, C. L., Wang, H. Y., "MTAD-TF: Multivariate time series anomaly detection using the combination of temporal pattern and feature pattern", Complexity, 2020: 1-9, (2020).

[22] Shih, S.Y., Sun, F.K., Lee, H.Y., "Temporal pattern attention for multivariate time series forecasting", Machine Learning, 108: 1421-1441, (2019).

[23] Che, Z., Purushotham, S., Cho, K., Sontag, D., Liu, Y., "Recurrent neural networks for multivariate time series with missing values", Scientific Reports, 8(1): 6085, (2018).

[24] Dai, G., Ma, C., Xu, X., "Short-term traffic flow prediction method for urban road sections based on space–time analysis and GRU", IEEE Access, 7: 143025-143035, (2019).

[25] Weerakody, P.B., Wong, K.W., Wang, G., Ela, W., "A review of irregular time series data handling with gated recurrent neural networks", Neurocomputing, 441: 161-178, (2021).

[26] Cook, A.A., Mısırlı, G., Fan, Z., "Anomaly detection for IoT time-series data: A survey", IEEE Internet of Things Journal, 7(7): 6481-6494, (2019).