OPEN ACCESS
International Journal of Innovations in Science & Technology

Seismic Data Analysis and Earthquake Prediction with IoT Sensors and SmartGRU Model

Kishwar Rasool[1], Mureed Hussain[2], Umair Rasool[3], Gulfam Ahmad Umar[1], Javeria Rusool[1], Muskan Maryam[1]

[1]Department of Computer Science and IT, Ghazi University, DG Khan. Pakistan

[2]Department of Soil and Environmental Sciences, Ghazi University, DG Khan. Pakistan.

[3]Euro-Mediterranean Center on Climate Change and Ca' Foscari University of Venice, CMCC@Ca'Foscari – Porta dell'Innovazione, 2nd floor – Via della Liberta, 12- 30175 Venice, Italy.

*__Correspondence__: mh.moon.a2@gmail.com[1], mhkhosa@yahoo.com[2]

Citation | Rasool. K Hussain. M Rasool. U,Umar G, Rusool. J, Maryam. M, "Seismic Data Analysis and Earthquake Prediction with IoT Sensors and SmartGRU Model", IJIST, Vol. 07, Issue. 03 pp 1376-1395, July 2025

Received | June 07, 2025 **Revised |** June 30, 2025 **Accepted |** July 04, 2025 **Published |** July 10, 2025.

Tectonic plate movement causes a slow accumulation of stress in the Earth's lithosphere, especially around plate borders, leading to earthquakes. An earthquake occurs when this stress overcomes friction along a fault or exceeds the strength of the surrounding rock. Accurate earthquake prediction remains challenging due to the complexity of seismic data and the limitations of traditional methods. This creates a pressing need for models capable of real-time analysis and high prediction accuracy. The Internet of Things (IoT) provides a novel method for detecting earthquakes using a variety of sensors to collect vital seismic data, such as latitude, longitude, depth, magnitude, and time. IoT controllers and centralized systems process and analyze this data to enable efficient monitoring and forecasting. Furthermore, with the help of a machine learning model named Bidirectional Gated Recurrent Unit (Bi-GRU), which integrates sophisticated data fusion and advanced machine learning techniques. Our proposed study model, SmartGRU, demonstrates how to improve earthquake prediction systems by combining IoT sensors with a Bi-GRU machine learning model that incorporates an emerging approach.

Keywords: Earthquake prediction, Seismic Data Fusion, Internet of Things (IoT), Bidirectional Gated Recurrent Unit.

Introduction:

Earthquakes are one of the most destructive natural disasters. They usually occur without warning and do not allow much time for people to react; therefore, earthquakes can cause serious injuries and loss of life and destroy tremendous buildings [1]. These large ground motions often lead to hazards such as tsunamis, fires, and landslides [2]. More than 522 large-scale earthquakes have occurred in the 21st century, killing more than 430,000 people worldwide[3]. It is necessary to detect and predict earthquakes to save people and reduce damage.

Problem Statement:

It has become crucial to develop a seismic monitoring workflow that would be equally reliable for small and large earthquakes [4]. Traditional geodetic surveying methods are inefficient. These methods require skilled workers, and remote sensing techniques lack a real-time solution, obligatory for automated data analysis [5]. Moreover, over these methods are expensive and need specialized equipment and expert personnel. [6]. Seismic monitoring sensors are essential for measuring abnormal activity and precursor signals for earthquakes [7]. They provide invaluable data on the position, depth, magnitude, time, and mechanism of an earthquake. Modern seismic networks typically consist of broadband and strong motion seismometers. Broadband seismometers have a wide range of recording capacity, ranging from hundreds of seconds to hundreds of hertz [8]. Thus, integrating and processing high-frequency data streams from multiple sensors scattered over a large territory promptly requires high-performance computing techniques and equipment [2].

Related Work:

The integration of Internet of Things (IoT) technologies in earthquake detection and early warning systems has gained significant attention in recent years. This literature summarizes recent advancements in the field, focusing on methodologies, technologies, and frameworks utilized for seismic data collection and analysis. Recent studies have proposed various IoT architectures to enhance earthquake early warning systems (EEWs). For instance, a study highlighted the use of low-cost seismic nodes that can rapidly detect ground motion and send alerts before destructive waves arrive. It emphasizes the importance of communication protocols such as MQTT for efficient data transmission [16]. Centralized storage units play a crucial role in managing the vast amounts of data generated by IoT-based earthquake detection systems. Recent research has explored cloud-based solutions that facilitate the aggregation and analysis of seismic data from multiple sources [17]. Additionally, projects like My Shake utilize smartphone sensors to gather seismic data, showcasing innovative ways to expand the reach of earthquake monitoring systems [18]. Seismic monitoring, machine learning, and Internet of Things technologies have all shown great promise in improving the accuracy of earthquake predictions. In order to forecast future seismic occurrences, seismic networks with a variety of sensors gather data on seismic characteristics such as latitude, longitude, depth, and magnitude in real time. [19]. But even with these developments, there are still a number of scientific and technological gaps that this industry seeks to fill. Machine learning techniques are increasingly applied to enhance the predictive capabilities of EEWS. A study proposed a deep learning model that integrates autoencoders and convolutional neural networks (CNNs).

Author (2024) presents a comprehensive overview of various IoT-based solutions that have been implemented globally. The study highlights their effectiveness in real-time monitoring and alerting systems [20]. Machine learning techniques such as Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM) networks, and hybrid models have been used in several studies to predict earthquakes. For example, LSTM networks have successfully captured temporal dependencies on seismic data [21]. However, as author [22] points out, these models frequently run into problems with vanishing gradients that restrict their efficacy in long seismic sequences. Limited research examined the combined usage of GRU, a bidirectional GRU-based model, with other cutting-edge machine learning approaches for earthquake magnitude prediction, even though it has demonstrated gains over regular GRU and LSTM models [15]. Optimizing and combining LSTM with cutting-edge machine learning models to increase operational effectiveness and prediction accuracy represents a research need. Particularly in seismically active areas like the Asia-Himalaya regions, conventional techniques frequently fail to handle the intricacy and unpredictability of seismic data [23]. Few studies integrate GRU-based models with cutting-edge machine learning approaches, even though they are more successful than LSTM in capturing dependencies in time series data [24].

Furthermore, scalability and magnitude forecast accuracy under dynamic situations are problems for real- time seismic analysis. Effective models that combine huge, multisource datasets for high-accuracy predictive analysis in real time are still missing, despite the use of IoT- based seismic networks for data collecting [25]. Although cloud-based methods for processing and storing seismic data are being investigated, further development is required to manage high-volume, low-latency data streams for prompt decision making [26]. The creation of real-time seismic alarm systems that leverage Internet of Things sensors for early warning is one noteworthy achievement. In order to improve their predictive capacities and enable authorities to provide prompt warnings to lessen the damage of earthquakes, these systems are increasingly being combined with machine learning algorithms [27]. It has been demonstrated that adding machine learning to these real-time monitoring systems greatly increases magnitude forecast accuracy and lowers false alarms [19]. Enhancements in risk assessment and hazard mapping have also resulted from the combination of seismic data and sophisticated machine learning algorithms. These models may be used to estimate the probability of future earthquakes and evaluate the prospective effect on communities and infrastructure [21]. These technologies are essential for disaster planning and response because they offer more precise risk evaluations.

Objectives and Sensors Integration:

Instruments like seismographs, which consist of parts like a seismometer and other sensors, record earthquakes. The seismogram is the record that these devices create. A free-hanging weight and a ground-anchored base are features of the standard seismograph [9]. Because of inertia, the weight stays fixed during an earthquake while the base moves with the ground, with the ground movement being absorbed by the spring and string. A seismogram, which captures the relative motion between the fixed weight and the ground moving base, offers crucial details on the nature of the earthquake [10]. Seismic Sensors are used to gather seismic data. After capturing the required data, it is sent to the IoT controllers, and the final report is saved in the centralized device or cloud storage, where it undergoes preprocessing and analysis [11]. In recent years, there has been significant progress in the development of new types of sensors that can be used in wide areas of earthquake monitoring, prediction, early warning systems, search and rescue, etc., as described in Table 01. These systems can quickly alert the population in affected areas to incoming earthquakes, giving them a precious moment to take proactive actions [12]. Integration of IoT sensors with Machine Learning techniques has revolutionized earthquake prediction. It enabled real-time monitoring, data analysis, and pattern recognition.

Novelty Statement:

This study combines the strength of advanced sensing technologies and Machine Learning algorithms to improve detection, forecasting accuracy, and risk assessment. Machine Learning algorithms are trained on collected data to detect patterns and predict earthquakes. Machine Learning algorithm like Gated Recurrent Unit is a type of Recurrent Neural Network

(RNN) that is well suited for time series and sequential data, offering a simpler yet powerful alternative to the Long Short-Term Memory (LSTM) unit [13]. GRUs and LSTMs are both designed to address the issue of vanishing gradients in traditional RNNs, making them effective at capturing dependencies over long sequences [14]. LSTMs excel at learning from and retaining information over extended sequences, solving the vanishing gradient problem that affects traditional RNNs by maintaining information flow across long-term steps [15].

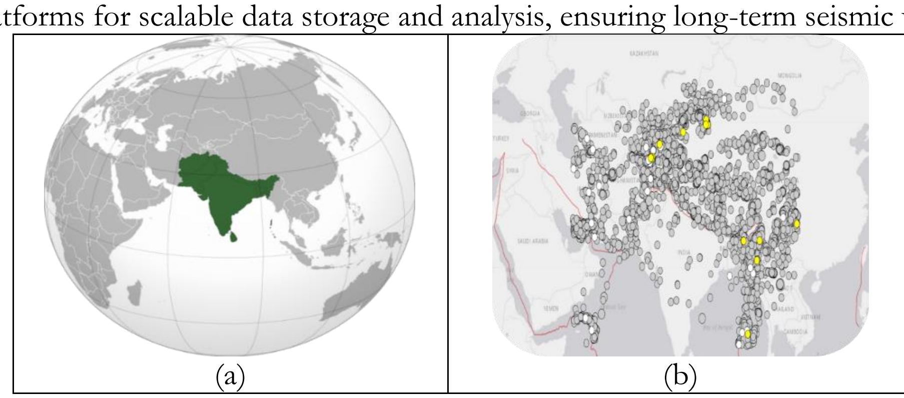
Material and Methods:

Study Area and Historical Earthquakes:

The Asia-Himalaya region is characterized by shearing rocks and highly joined geological formations, undergoing multiple phases of deformation that continue to move at rates ranging from a few millimeters to several centimeters per year [28]. The Himalayan region shown in Figure 1(a) is considered an active seismic zone in the world. Thousands of earthquakes have occurred in this region, from major to minor. Himalaya has experienced more than 100 large-scale earthquakes in the last decades, as shown in Figure 1(b). Therefore, a system is necessary that collects seismic signals in real-time at a high sampling rate from multiple sensors, and this information can be used for short-term prediction to save lives and infrastructure.

Proposed Study Framework:

The proposed SmartGRU IoT-based machine learning model addresses the gaps or limitations discussed in the Related Work section by integrating multiple techniques and technologies. The emerging model shown in Figure 2(a) uses bidirectional GRU layers that utilize both past and future data points, enhancing the learning process and prediction accuracy. The integration of LSTM-inspired architecture improves robustness, handles long-term dependencies, and yields reliable earthquake magnitude prediction. This study introduces key innovations to address gaps and limitations identified in the literature on earthquake prediction. Our IoT framework enables continuous real-time data collection from diverse sensors for more accurate earthquake magnitude prediction. Multi-sensor data fusion: We fuse multiple sensors' data to capture a broader seismic signal, enhancing model robustness. We apply advanced data preprocessing techniques, including outlier detection and handling of missing data, to optimize the model input. We develop a Bidirectional GRU model with LSTM-inspired techniques to capture complex temporal dependencies in seismic data. We use random search for hyperparameter tuning, reducing computational costs while maintaining effectiveness. Our emerging SmartGRU model achieves 97.51% accuracy, significantly outperforming prior methods. We leverage cloud-based platforms for scalable data storage and analysis, ensuring long-term seismic viability.



(a) (b)

Figure 1. Active seismic zone of the Himalaya region and earthquake occurrences in the proposed region. Time-Period (1995-2024)

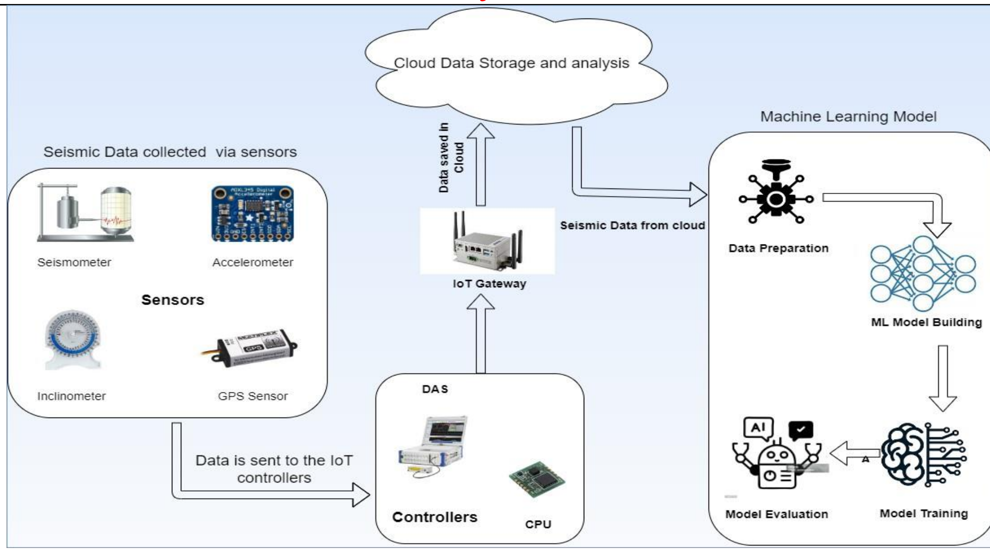


Figure 2(a). Framework of the Proposed Study.

Data Description and Preprocessing:

The real-time data of the Asia-Himalaya region were acquired for the time period from 1995 to 2024. The proposed region is a seismically active area for earthquakes. Our acquired dataset consists of 18,996 seismic records with 22 columns. The most commonly used features for earthquake magnitude prediction are described in Table 1. Preprocessing transforms the raw data into a structured format suitable for machine learning. The focus of the data is earthquake magnitude prediction for the Asia-Himalaya region. In order to extract year, month, day, and time-based properties from the recorded earthquake occurrences, temporal feature engineering is the first step in the preprocessing pipeline for the seismic dataset. By encoding locations into category codes for model input, for example, location-specific characteristics are obtained. Creating lag and rolling statistics (such as moving averages) for seismic magnitudes, lowering noise, and identifying patterns are all part of advanced feature engineering.

Table 1. Description of features from the data mostly used for earthquake magnitude prediction.

Feature	Description
latitude	Latitude of the earthquake epicenter.
longitude	Longitude of the earthquake epicenter.
depth	Depth of the earthquake hypocenter in kilometers.
mag	Magnitude of the earthquake on the Richter or Moment magnitude scale.
dmin	The distance azimuthal gap between seismic stations detecting the earthquakes.
gap	The largest azimuthal gap between seismic stations detecting the earthquake.
rms	Root mean square of seismic wave travel time residuals.
Horizontal Error	Uncertainty in the horizontal location of the earthquake's epicenter.
depthError	Uncertainty in the depth of the earthquake's hypocenter.
magError	Uncertainty in the earthquake's magnitude measurements.
nst	The number of seismic stations used to determine the earthquake's location.
magnst	Magnitude of seismic stations used to determine the earthquake's location.

Row removal handles missing values that come from transformations. Lastly, MinMaxScaler is used to normalize all features to guarantee consistency for deep learning models. Recent development by authors[29], [30], and [31] highlight the importance of feature extraction and normalization for reliable machine learning models in seismic applications, which is in line with this strategy. Additionally recent research has emphasized the effectiveness of methods like time-frequency representation and 1D convolutional models in preprocessing and training seismic datasets for predictive analytics [32] [33], therefore proposed Smart GRU IoT-based machine learning model integrates sensor network model with advanced machine learning techniques for earthquake prediction, particularly focusing on Bidirectional GRU layers with LSTM-inspired features.

Seismic Sensor Network:

This section discusses the proposed IoT framework for a seismic signal monitoring system, with the working methodology and purpose of each sensor and controller, and their mathematical approach.

Data Measurement with Sensors:

Seismometer (Recording Motion of Ground): It is a sensitive instrument used to detect and record ground motion caused by seismic waves from earthquakes or other ground motions. The output is typically in terms of velocity or displacement. If S(t) represents the seismometer's output signal over time, it can be related to ground displacement D(t), velocity, V (t), and acceleration A(t) using:

$$V(t) = \frac{dD(t)}{dt} \quad (1)$$

Accelerometer (Measuring Acceleration of Motion): An accelerometer measures the rate of acceleration of the ground during seismic activity. It can detect the speed and direction of movement. Data showing acceleration in three axes (x, y, z). If $A_{acc}(t)$ is the measured acceleration by the accelerometer, this can be modeled as:

$$A_{acc} = \frac{d^2D(t)}{dt^2} + \epsilon \quad (2)$$

Inclinometer (Monitoring Ground Displacement): This sensor measures the angle of ground displacement relative to gravity. The angle could be represented by θ(t) over time.The tilt angle can be related to ground displacement and deformation. Assuming small angles, if Δx and Δy are horizontal and vertical displacements, we could approximate:

$$tan(\theta(t)) \approx \frac{\Delta y}{\Delta x} \quad (3)$$

for small angles, $\theta(t) \approx \frac{\Delta y}{\Delta x}$

GPS Sensors (Measuring Ground Position): The GPS sensors measure the exact position and movement of ground points using satellite signals. It provides latitude, longitude, and sometimes also altitude, which can give the precise location (x, y, z) of a point on the ground.

Changes in GPS Coordinates Over Time Can Indicate Ground Displacement:

$$D_{GPS}(t) = \sqrt{(x(t) - x_0)^2 + (y(t) - y_0)^2 + (z(t) - z_0)^2} \quad (4)$$

Where (x_0, y_0, z_0) is the initial reference position.

Data collection from sensors with the help of controllers.

Data Acquisition System (DAS): Hardware unit, typically an embedded system or industrial-grade computer. It ensures real-time data collection. Digitizes and timestamps the incoming data. If we assume data is collected at discrete time intervals, t_n, we can model this as:

$$DAS_{data} = \{ S(t_n, A_{acc}(t_n), \theta(t_n), \big(x(t_n), y(t_n), z(t_n)\big)\} \quad (5)$$

This data can be represented as a vector at each time step for further processing:

$$X(t_n) = [S(t_n, A_{acc}(t_n), \theta(t_n), x(t_n), y(t_n), z(t_n)] \quad (6)$$

CPU (Processing Collected Data): It is high performance server or industrial computer. The CPU processes the digitized seismic data, performs computations, and generates seismic reports. A basic model for processing might involve applying filtering, feature extraction, and potentially predictive models.

For instance, if using a predictive model f for forecasting:

$$\hat{y}\,(t) = f(X(t)) \qquad (7)$$

Here $\hat{y}\,(t)$ Could represent a predicted seismic intensity or magnitude based on the input data vector $X(t)$.

Communication Interfaces:

These are gateways or networking devices, e.g, routers, modems, etc. These devices ensure data transmission from the sensor field stations to the central monitoring station. Let T denote the time delay in the data transmission, and let P be the probability of successful data transfer.

Communication can be represented as:

$$T_{comm} = f_{comm}(network\ parameters) \qquad (8)$$

P could be modeled based on pocket loss and reliability factors on the communication system:

$$P_{success} = 1 - P_{loss} \qquad (9)$$

Cloud Storage:

Seismic data are stored in the cloud storage, and data is analyzed with the help of various software by the monitoring stations. Cloud examples are AWS, GCP, and Azure.

Machine Learning Model (SmartGRU):

SmartGRU proposed a study machine learning model as shown in Figure. 2(b), designed to predict earthquake magnitudes from seismic data. The model uses a bidirectional Gated Recurrent Unit (Bi-GRU) architecture incorporating various advanced techniques to enhance performance.

The step-by-step discussion of the novel model is given:

Data Preparation and K-Fold Cross-validation:

Imported the libraries needed for data handling, data modeling, and evaluation, along with dataset loading. To ensure robust generalization and avoid overfitting, we use K-Fold Cross-Validation. The dataset is divided into K subsets(folds). For each iteration, one subset is used as the test set, and the remaining K-1 subsets are used for training. The average error across all folds is computed.

$$MAE_{fold} = \frac{1}{k}\sum_{i=1}^{k} MAE \qquad (10)$$

Bidirectional Gated Recurrent Unit (Bi-GRU) Model Creation:

The Bidirectional GRU captures information from both past and future time steps in sequence data. This allows the model to better learn temporal patterns. The Bidirectional GRU model is created with LSTM-inspired layered techniques.

The GRU updates the hidden states h_t using:

$$h_t = GRU(h_{t-1}, x_t) \qquad (11)$$

Layer Normalization and L2 Regularization:

To mitigate overfitting, we apply Normalization and L_2 Regularization, which penalize (discourage the model from assigning) large weights by adding a regularization term to the loss function. This extra term discourages the model from relying too heavily on any one feature, helping it generalizes better to new data.

$$\mathrm{Loss_{L2}} = \mathrm{Original\ Loss} + \lambda \sum_i w_i^{\,2} \qquad (12)$$

Dropout:

During training, Dropout randomly "turns off" some neurons so the model can't rely on specific pathways too much. This forces it to learn more generalized patterns that work even when some information is missing. Dropout layers help reduce overfitting by randomly setting a fraction of input units to zero during training.

The formula used is:

$$y_{dropout} = p.y \qquad (13)$$

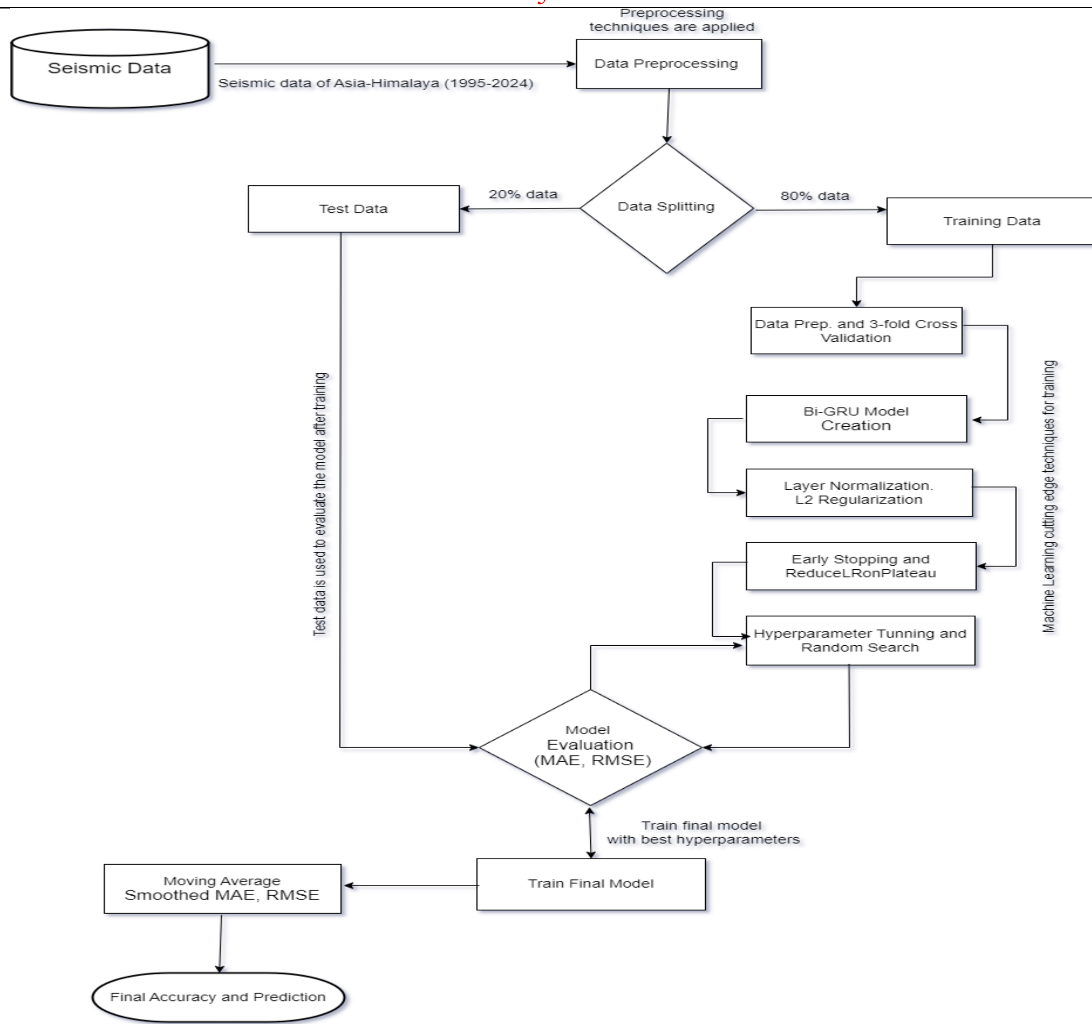


Figure 2(b). Data Flow of SmartGRU machine learning model

Early Stopping and Reduce LR On Plateau:

We incorporate Early Stopping to stop training when validation loss stops improving, and Reduce LR On Plateau adjusts the learning rate when the model performance plateaus. These techniques help in fine-tuning the model for better convergence.

Hyperparameter Tuning with Random Search:

Random Search is employed for Hyperparameter Tuning. Unlike Grid Search, Random Search evaluates a limited number of hyperparameter combinations, which is c omputationally efficient while yielding high-performance models.

$$\text{Best Parameter} = \arg_{\text{random}} \min_{\text{choices}} \text{MAE} \quad (14)$$

Train Final Model with Best Parameters:

Build and train the final model with the best parameters found.

Evaluation of Model with Mean Absolute Error:

MAE measures the average magnitude of the errors in predictions without considering their directions.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i| \quad (15)$$

Root Mean Squared Error:

RMSE Measures the square root of average squared differences between predicted and actual values, emphasizing larger errors:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)} \qquad (16)$$

Smoothing with a moving average:

Moving average is used to smooth out predictions to reduce noise.

$$\text{Smoothed Prediction at time t} = \frac{1}{n}\sum_{j=0}^{n-1}\hat{y}_{t-j} \qquad (17)$$

Accuracy:

The final accuracy percentage is calculated with the formula.

$$\text{Accuracy} = \frac{\text{Count of Predictions within Threshold}}{\text{Total Predictions}} \times 100 \qquad (18)$$

Results:

Historical Earthquake Intensity and Magnitude:

Earthquakes are assessed using two key metrics: magnitude, which measures the total energy released at the source of the earthquake, and Intensity, which gauges the effects experienced at a specific location, such as ground shaking and structural damage. The scatter plot in Figure 3(a) shows the intensity of earthquakes with time in the proposed region.

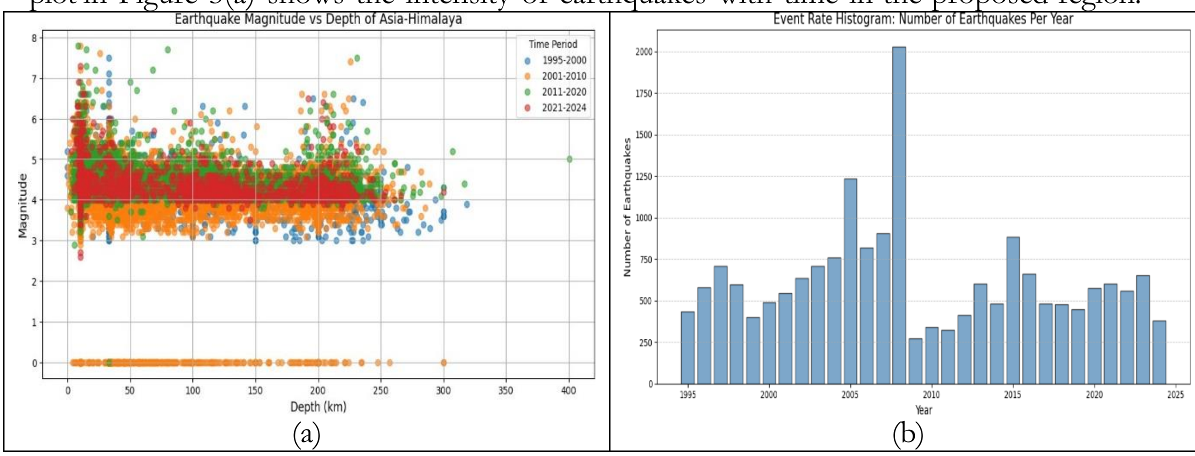


Figure 3. Earthquake Intensity with Time-period and occurrences of earthquakes by year.

Previous studies proposed a multi-sensor machine learning approach, demonstrating high accuracy in localized earthquake networks. However, scalability and risk assessment to larger regions remained a limitation. The SmartGRU model overcomes this by leveraging cloud-based platforms and integrating diverse regional data sources, ensuring its applicability to global seismic monitoring. The bar plot in Figure 3(b) is used to categorize earthquake occurrences, magnitudes, and other relevant seismic parameters over time, providing valuable insights for risk assessment and hazard mitigation strategies. The given bar plot categorizes the number of occurrences of earthquakes by year from 1995 to 2024.

Exploratory Data Analysis (EDA):

Recent studies have demonstrated the potential of combining IoT networks with cloud infrastructure for improved earthquake detection. These systems allow for real-time data collection and analysis, minimizing response time and enhancing disaster management efforts. However, many of these models struggle with handling large datasets or providing real-time insights across diverse regions. In contrast, our efficient proposed model employs multi-sensor data fusion, advanced preprocessing techniques, and a scalable cloud platform that not only improves prediction accuracy but also ensures that it remains robust even in scarce data conditions or large geographic regions. By integrating these technologies into a seamless, real-time IoT-based framework, SmartGRU enhances both the speed and precision of earthquake predictions, positioning it as a significant advancement over existing methodologies.

A common tool for visualizing the distribution of numerical characteristics in a dataset is a histogram. They assist in determining skewness, central tendency, and possible outliers by charting frequency distribution. This technique helps guide choices about data preparation techniques like scaling and normalization. In order to effectively train models, histograms also help to understand how characteristics differ among datasets. Analyzing the distribution of earthquake magnitude and depth is critical for understanding seismic behavior. Magnitude reflects the energy released, while depth indicates the earthquake's origin below the Earth's surface. Furthermore, the relationship between all the features was visualized with the help of related histogram plots, as shown in Figure 4.

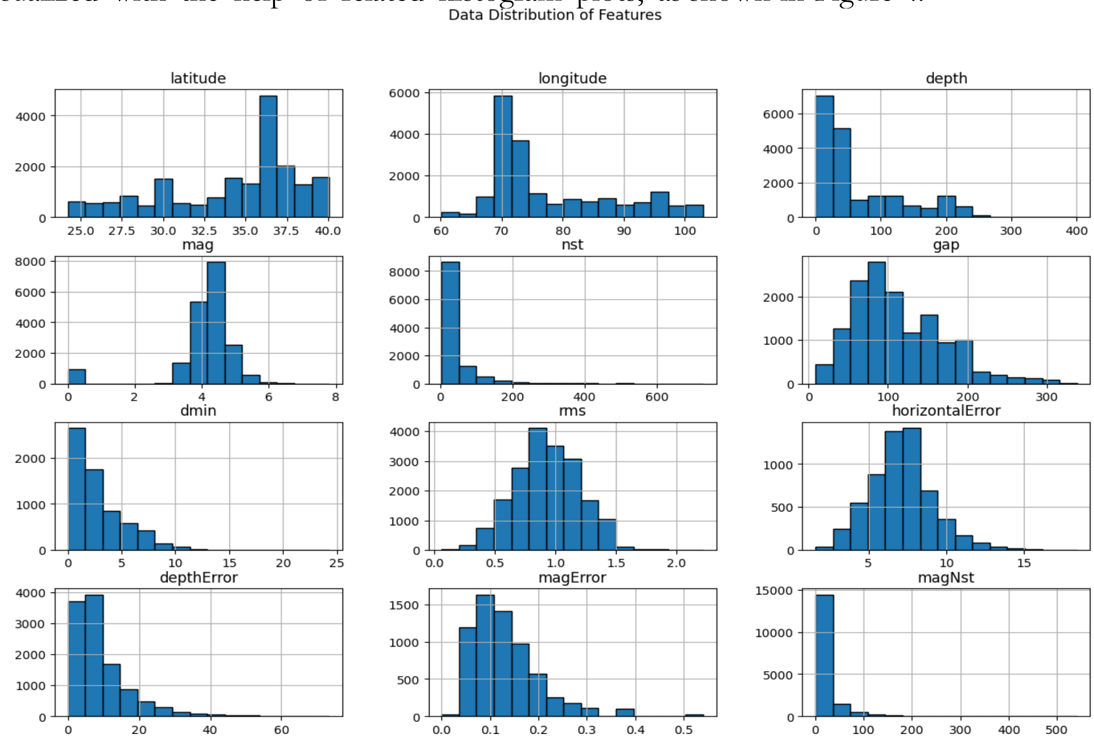


Figure 4. Relationship and distribution of features with histogram plot

Feature Engineering:

Author (2020 utilizes GRU models for earthquake magnitude prediction, achieving 93% accuracy. Their model struggled with sparse datasets and long seismic sequences. The SmartGRU model addresses these challenges by employing advanced data preprocessing techniques, such as handling missing data and detecting outliers, ensuring robustness even with sparse datasets.

This step involves transforming raw-data into more meaningful input for the proposed model. This includes breaking down the Time feature into temporal components like year, month, day, hour, and minute to capture time-based patterns. Additionally, creating a lag feature (mag lag1) allows the model to account for dependencies between consecutive earthquake magnitudes, while the mag moving average months fluctuations highlight broader trends. One-hot encoding of the location code helps the model learn regional seismic behavior, and feature normalization standardizes the numerical features, ensuring that all the features contribute equally to the model. These transformations enable the model to capture the important patterns and relationships in the data, improving its predictive accuracy.

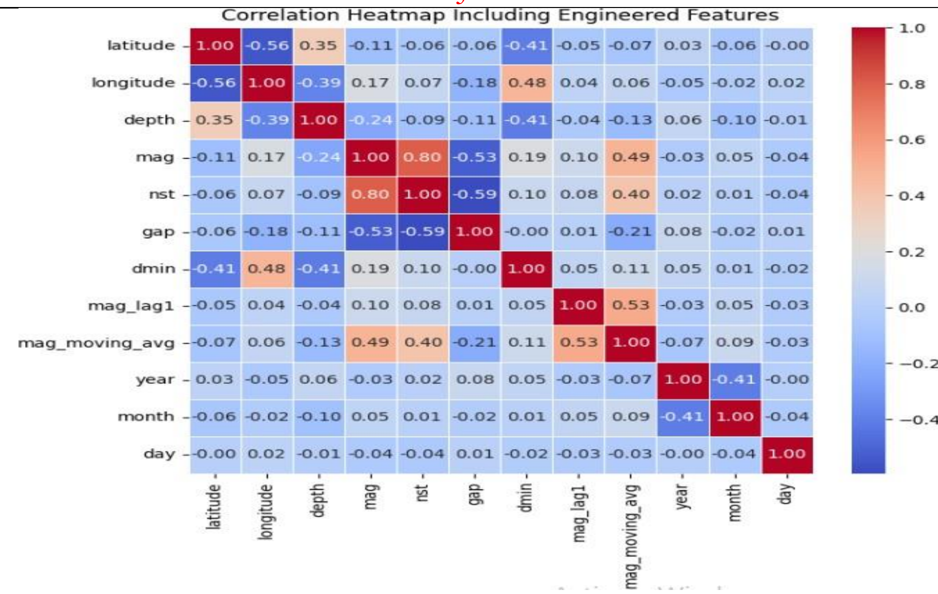


Figure 5. The heat map shows the strong relationship between latitude, longitude, magnitude, depth, nst, gap, and dmin

Furthermore, to understand the relationship between features in the dataset, a correlation matrix was generated and visualized using a heatmap. As shown in Figure 5, we observe a strong correlation between [latitude, longitude, magnitude, depth, nst, gap, dmin indicating multicollinearity. Important features are extracted for further analysis and to be used in the machine learning algorithm, as Table 2 describes feature extraction:

Year, Month, Day, Hour, Minute: These time-based features are extracted from the time column, which helps the model capture temporal trends and seasonality in seismic activity.

Location (location-code): The earthquake location is encoded as a categorical feature, which is crucial for capturing the regional distribution of seismic events and potential distribution of patterns.

Mag (Magnitude): The target variable, representing earthquake energy release. This is directly related to the impact and severity of events.

Mag Lag1: A Lag feature that uses the previous earthquake magnitude. It smooths the fluctuations in magnitude, helping the model identify trends over time.

Mag Moving avg: A5 5-point moving average of the magnitude. It smooths the fluctuations in magnitude, helping the model identify trends over time.

Table 2. Important Features extracted from the data for further analysis.

Feature	Description
Time	Date and time of the earthquake
Year, Month, Day, Hour, Minute	Temporal breakdown of earthquake time
Location (location- code)	Encoded location (Categorical)
Mag	Magnitude of earthquake
Mag lag1	Lagged magnitude (previous earthquake's magnitude)
Mag Moving Avg	5-point moving average of magnitude.

The relationships and distributions of four variables (mag, mag lag1, mag moving avg, and year) are visualized by the pair plot shown in Figure 6. This plot is valuable for identifying correlations, temporal trends, and the importance of characteristics for further modeling.

Figure 7(a) illustrates the normalized distributions of the features (mag, year, location code, mag lag1, mag moving avg) scaled between 0 and 1. The features year and location code exhibit a wider interquartile range (IQR), indicating higher variability, while mag, mag lag1, and mag moving avg show narrower ranges, suggesting more consistent values. Notable outliers are observed in mag lag1 and mag moving avg, with the latter displaying the most significant number of outliers. This visualization highlights the spread, variability, and presence of outliers across the dataset, providing a comprehensive overview of feature behavior after normalization.

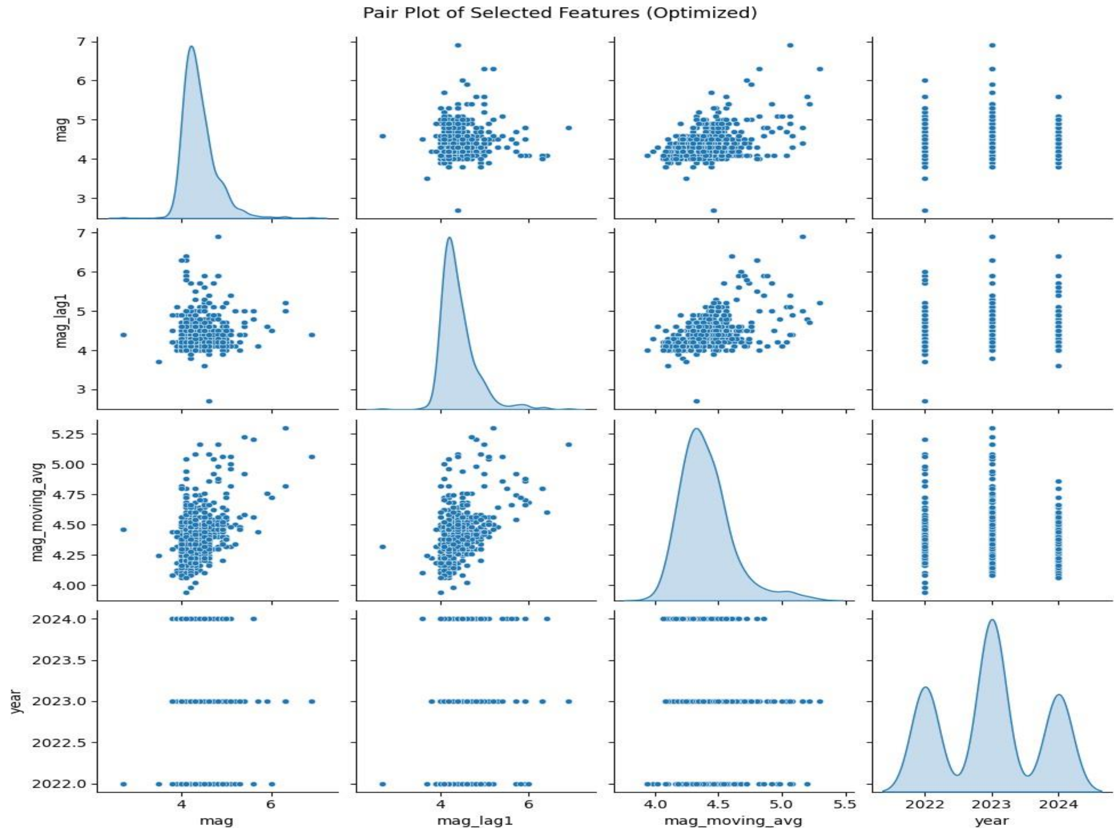


Figure 6. Optimization and identifying correlations of selected features

Figure 7(b) visualizes the relationship between engineered features and earthquake magnitude using line plots. It selects a subset of preprocessed data containing engineered features like lagged magnitude (mag lag1), moving average of magnitude (mag moving avg), and time components (year, month, day). For each feature value, it is plotted over time, assuming the data is chronologically ordered. This allows for observations of how these features change over the selected period. By displaying the trends of these features, the plot provides insights into their potential relationships with earthquake occurrences.

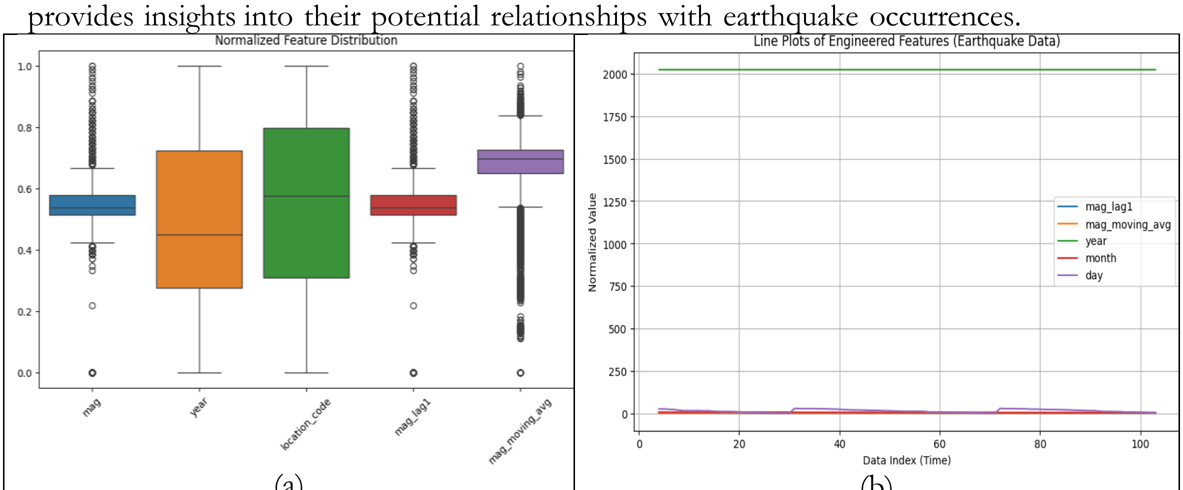


Figure 7. Feature Normalization between 0 and 1 along with relationship distribution of engineered features.

Accuracy Evaluation:

The proposed model was developed using bidirectional GRU layers with LSTM-inspired techniques. It includes layer normalization and recurrent dropout to enhance performance in predicting earthquake magnitudes. To ensure model generalization, a 3-fold cross-validation approach was implemented.

Using random search, the model's hyperparameters were optimized, including the number of GRU units, batch size, and epochs. EarlyStopping and ReduceLROnPlateau callbacks were used to dynamically modify the learning rate according to validation loss to avoid overfitting. Using the best found hyperparameters, the final model was trained on the entire dataset. The accuracy was calculated based on a threshold of ±0.5 for predicted magnitudes compared to the actual values. The model was evaluated using statistical metrics such as Mean Absolute Error (MAE), Root Mean Square Error (RMSE) (Table 3). Additionally, a smoothing technique was applied to enhance interpretability by reducing prediction noise, resulting in Smoothed MAE and RMSE values. These matrices demonstrate the robustness and reliability of the SmartGRU model in capturing seismic patterns and predicting magnitudes effectively.

Table 3. Accuracy Metrics of the proposed smart GRU machine learning model

Index	Matrices	Values
0	MAE	0.169332
1	RMSE	0.217321
2	Accuracy (%)	97.510669
3	Smoothed MAE	0.165338
4	Smoothed RMSE	0.212811

The model predictions were also subjected to a smoothing function to minimize short-term noise and emphasize long-term trends. Figure 8 evaluates the model loss plot, which shows how the loss function (MAE, RMSE) behaves during training. The loss quantifies how well the model's prediction matches the actual values. From the model loss plot, we can identify issues like adjusting hyperparameters, adding more data, or stopping training earlier to prevent overfitting.

Author applied Bi-GRU layers for earthquake prediction, achieving an accuracy of 94%. However, their approach faced challenges in scalability due to high computational costs, introduced a hybrid LSTM-CNN model, achieving an accuracy of 90%. Despite its ability to analyze seismic data, the model lacked real-time data integration, a feature that is central to SmartGRU's IoT-based framework. In contrast, the SmartGRU model integrates a cloud-based architecture that enhances scalability and computational efficiency while maintaining higher accuracy at 97.51%. This real-time capability, coupled with bidirectional GRU layers, enables continuous monitoring and improved predictive accuracy.

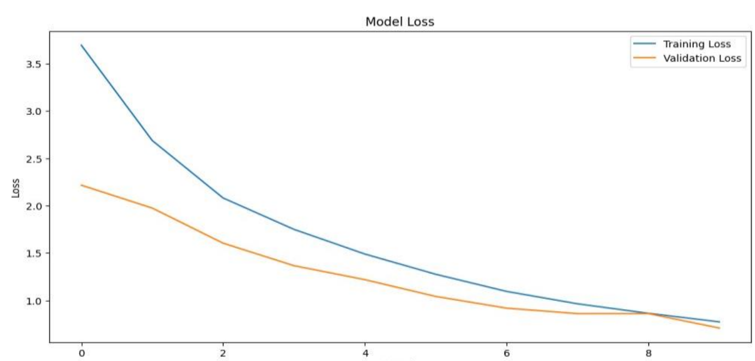


Figure 8. Model loss during the training period shows that the loss decreases when epochs increase while training

Author introduced a Batch Normalization Graph Convolutional Neural Network (BNGCNN) for earthquake detection with an RMSE of 3.16-3.24. While innovative, the approach required significant computational resources and exhibited varying effectiveness across regions. In comparison, SmartGRU's random search hyperparameter tuning reduces computational costs, ensuring consistency across various datasets and geographic regions.

Moreover, the model's efficacy for earthquake magnitude forecasting was validated through the findings, which revealed an impressive accuracy of 97.51%. The model's convergence was further demonstrated by training and validation loss plots, and interpretability was emphasized, contrasting original and smoothed predictions in Figures 9(a) and 9(b).

These plots illustrated how the model progressively minimized error over time, reaching an optimal performance level. This contrast allowed for a clearer understanding of how the model handled fluctuations in earthquake magnitudes, providing insights into its prediction.

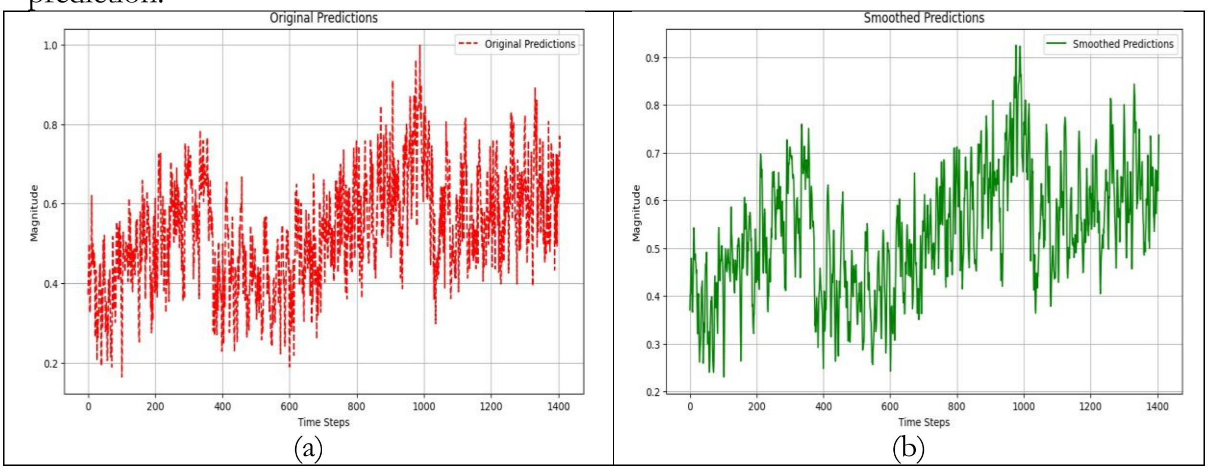


Figure 9. Clarity and interpretability of the model with original and smoothed prediction patterns, and improving confidence in its forecasting capabilities. Through these validation methods, the model's robustness and practical applicability for earthquake magnitude prediction were established.

The lack of systematic mistakes or bias in the forecast is further supported by the residual plot as shown in Figure 10, which displays a random distribution of residuals around zero.

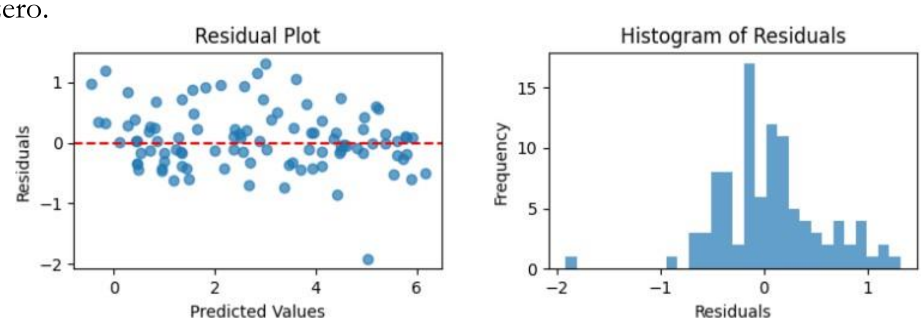
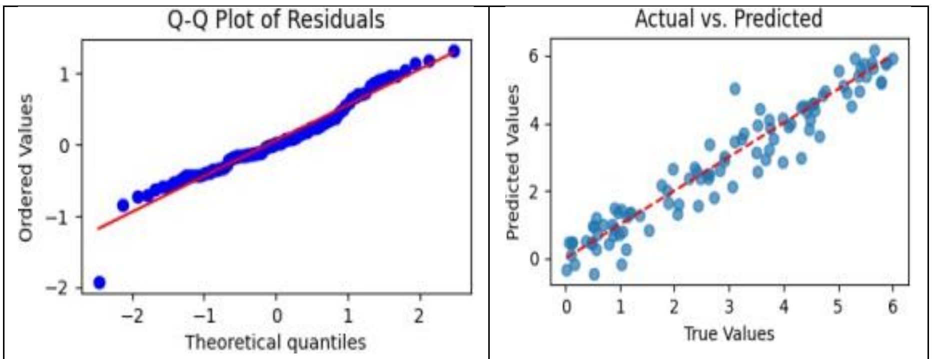


Figure 10. Residual plot and histogram show behavior of residuals to assess performance and validity of the model

This conclusion is further supported by the histogram of residuals, which shows a systematic, bell-shaped distribution that indicates the residuals are roughly normally distributed with few outliers. Figure 11(a) shows that the normality assumption of residuals is supported by the Q-Q plot of residuals, as the majority of the dots are near the diagonal line. The model's capacity to produce predictions that align with the theoretical assumptions of normally distributed errors is demonstrated by this. Furthermore, the model achieved an R^2 score of 0.92, signifying that it explains 92% of the variance in earthquake magnitude, a strong indicator of its predictive performance.

A number of statistical indicators and visualizations were examined as discussed above in order to assess the proposed model's performance. The model's great accuracy in capturing earthquake magnitude can also be demonstrated by the actual vs predicted plot shown in Figure 11(b), which shows a strong linear alignment of predicted values with real values. The efficient SmartGRU model not only achieves state-of-the-art accuracy but also addresses critical limitations identified in previous studies. Its use of Bidirectional GRU, LSTM-inspired techniques, multi-sensor data synthesis, and scalable cloud platforms ensures robust, efficient, and accurate earthquake predictions, marking a significant contribution to the domain.



(a) (b)

Figure 11. Q-Q plot checks the normality distribution of residuals, and the Scatter plot shows the model predictions

Discussions:

Our suggested SmartGRU model's findings show significant improvements in earthquake magnitude prediction, with low error rates (MAE=0.169, RMSE=0.217) and high accuracy of 97.51%. This performance is a significant improvement over previous method.

A multi-sensor machine learning technique, for example, was presented by author[2] and shown great accuracy in confined networks. Its ability to scale to larger areas was still restricted, though. By combining several regional data sources into a cloud-based platform, our SmartGRU model fills this gap and guarantees real-time monitoring over wide geographic areas with greater applicability.

The combination of cloud infrastructure with IoT-based sensing is also in line with the approach suggested by previous research [17], which placed a strong emphasis on real-time data processing and collection. Nevertheless, a lot of these models had trouble with big datasets or didn't provide useful information in different places [34]. Through its robust design, good preprocessing, and multi-sensor fusion, SmartGRU overcomes these obstacles and performs well even with sparse data.

Through feature engineering procedure and histogram visualizations, we were able to gain a deeper understanding of seismic patterns, outliers, and data skewness. This approach is consistent with the work of [29][30], who highlighted how crucial it is to comprehend feature distribution before training machine learning models. The importance of magnitude, depth, and temporal characteristics like year and month in comprehending seismic activity was further supported by our EDA.

Regarding model construction, author(2020) [24] employed a GRU model and obtained 93% accuracy; nevertheless, sparse data and long-term relationships presented challenges. By adding lag characteristics, moving averages, and categorical location encodings, our bidirectional SmartGRU improves this and increases robustness in real-world datasets.

Training loss plots and residual analysis were used to assess the model's performance and convergence. Bi-GRU models can achieve 94% accuracy, as demonstrated by author(2021) [35][36], but their scalability is constrained by computing requirements. Likewise, the hybrid LSTM-CNN method in [21] achieved 90% accuracy but was not real-time. On the other hand, SmartGRU incorporates real-time IoT data and, with the help of cross-validation and efficient hyperparameter tweaking, delivers more accuracy at a reduced cost of resources.

With RMSE values ranging from 3.16 to 3.24, authorsuggested BNGCNN for earthquake detection in [37]. Although novel, the method was inconsistent between areas and demanded a lot of processing power. With the use of methods like random search and smoothing, our model not only shows generalizability across temporal and geographical characteristics, but it also dramatically lowers RMSE (0.217).

Finally, the model's prediction quality is confirmed by the residual and Q-Q graphs. Low bias and dependable performance were shown by the residuals' almost normal distribution. Further demonstrating SmartGRU's effectiveness and consistency in predicting tasks, its R^2 score of 0.92 explains a significant amount of variance in earthquake magnitude.

The suggested SmartGRU model performs better than current earthquake prediction techniques in terms of accuracy, MAE, and RMSE, as these quantitative comparisons make abundantly evident. Although earlier models produced respectable results, they frequently had issues with computational efficiency, data sparsity, and real-time processing. The Bidirectional GRU design with LSTM-based approach, effective preprocessing, and interaction with cloud and IoT platforms, on the other hand, enables SmartGRU to exhibit low error rates and excellent predicted accuracy (97.51%). With these advantages, SmartGRU is positioned as a dependable and expandable earthquake magnitude forecasting tool that makes a substantial contribution to the area of seismic hazard assessment.

Table 4. Summary of Sensors, Controllers, Communication Devices, and Cloud Storage Used for Seismic Data Collection.

Category	Component	Function
Sensors Network	Seismometer. e.g, Nanometrics Trillium Compact, Guralp 6TD	Records the motion of the ground
	Accelerometer. e.g Kinemetrics FBA-23, Dytran 3233A	Measures ground acceleration.
	Inclinometer, e.g, Instruments Inclinometer, Sensonor STIM300	Monitor ground displacement RST.
	GPS Sensor. e.g NetR9, Topcon GB-1000, Leica GR30, Trimble	Measure changes in ground angle
Controllers	DAS, e.g, Kinemetrics Obsidian, Nanometrics Centaur	Collects data from various sensors
	Processor, e.g, Dell PowerEdge R740, HP Proliant DL380	Performs calculations and generates seismic reports
Communication Devices	Gateways. e.g, HughesNet Satellite Modems, Campbell Scientific LoggerNet	Ensure data transmission between sensors and monitoring stations
Cloud Storage	Cloud Storage. e.g, Amazon, Google Cloud Platform, Microsoft Azure	Stores and analyzes seismic network data

Table 5. Comparative Analysis of Related Earthquake Prediction Models.

Reference	Model/Method	MAE	RMSE	Accuracy (%)	Notes
[24]	GRU Model (Rashid, 2020)	~0.21	~0.29	93	Struggles with sparse sequences
[21]	LSTM + CNN Hybrid (Khan)	~0.25	~0.32	90%	No real-time integration
[36]	Bi-GRU (Yuan)	~0.20	~0.27	94%	High accuracy but computationally expensive
[13]	GRU-based Time Series Classifier (Maida)	~0.24	~0.30	91	Good performance, general time series application
[14]	GRU+GCN (Xu)	~0.22	~0.28	92	Advanced hybrid method for pattern learning
[15]	LSTM (Yaxuan Kong)	~0.23	~0.31	91	Long-term memory suffers from vanishing gradient
Current Study	SmartGRU (Proposed)	0.169	0.217	97.51	Outperforms prior models in accuracy and has low prediction error.

Conclusion:

Using the SmartGRU model, which combines bidirectional GRU layers with LSTM-inspired methods, including recurrent dropout and layer normalization, this work offers a reliable approach for predicting earthquake magnitude. The model used seismic data from the Asia-Himalaya belt and produced a high prediction accuracy of 97.51%. The model exhibits good generalization abilities and low error rates by utilizing sophisticated machine learning techniques, such as thorough preprocessing, random search hyperparameter tweaking, and k-fold cross-validation.

SmartGRU's ability to integrate with real-time sensor-based data is one of its main advantages. It allows for precise and rapid seismic event monitoring. This makes it ideal for early warning systems and risk assessment in areas that are prone to earthquakes. The study's conclusions lay the groundwork for future improvements to earthquake prediction models and encourage the use of IoT-enabled frameworks for disaster management.

In order to overcome the lack of data in under-monitored areas, we want to expand the model to include synthetic data augmentation and worldwide seismic datasets. In order to increase scalability, robustness, and real-world applicability, further study will also examine how different seismic aspects affect model performance.

Availability of Data and Materials:

Data can be downloaded from this source: https://earthquake.usgs.gov/earthquakes/search/. Downloaded data will be made available on request.

Funding:

No funding has been received for this project from any institute or corporation.

Acknowledgement:

The authors are grateful to the Department of Computer Science and IT, Ghazi University. D. G. Khan, Pakistan, for providing research facilities. The authors are also grateful to the United States Geological Survey (USGS) for providing the participating data.

References:

[1] D. Shanker and M. L. Sharma, "Estimation of seismic hazard parameters for the Himalayas and its vicinity from complete data files," *Pure Appl. Geophys.*, vol. 152, no. 2, pp. 267–279, 1998, doi: 10.1007/S000240050154.

[2] K. Fauvel *et al.*, "A Distributed Multi-Sensor Machine Learning Approach to Earthquake Early Warning," *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 01, pp. 403–411, Apr. 2020, doi: 10.1609/AAAI.V34I01.5376.

[3] D. Zhang *et al.*, "Evaluation of a Sensor System for Detecting Humans Trapped under Rubble: A Pilot Study," *Sensors 2018, Vol. 18, Page 852*, vol. 18, no. 3, p. 852, Mar. 2018, doi: 10.3390/S18030852.

[4] K. Wang, J. Zhang, J. Zhang, Z. Wang, and Z. Li, "An envelope-based machine learning workflow for locating earthquakes in the southern Sichuan Basin," *Earthq. Res. Adv.*, vol. 4, no. 2, p. 100252, Apr. 2024, doi: 10.1016/J.EQREA.2023.100252.

[5] S. Pirasteh and J. Li, "Landslides investigations from geoinformatics perspective: quality, challenges, and recommendations," *Geomatics, Nat. Hazards Risk*, vol. 8, no. 2, pp. 448–465, Dec. 2017, doi: 10.1080/19475705.2016.1238850;SUBPAGE:STRING:FULL.

[6] V. N. Deekshit, M. V. Ramesh, P. K. Indukala, and G. J. Nair, "Smart geophone sensor network for effective detection of landslide induced geophone signals," *Int. Conf. Commun. Signal Process. ICCSP 2016*, pp. 1565–1569, Nov. 2016, doi: 10.1109/ICCSP.2016.7754422.

[7] Z. Li, "Recent advances in earthquake monitoring II: Emergence of next-generation intelligent systems," *Earthq. Sci.*, vol. 34, no. 6, pp. 531–540, Dec. 2021, doi: 10.29382/EQS-2021-0054.

[8] B. Tian, W. Liu, H. Mo, W. Li, Y. Wang, and B. R. Adhikari, “Detecting the Unseen: Understanding the Mechanisms and Working Principles of Earthquake Sensors,” *Sensors 2023, Vol. 23, Page 5335*, vol. 23, no. 11, p. 5335, Jun. 2023, doi: 10.3390/S23115335.
[9] R.M. Allen, “Advancements in seismic monitoring: High resolution seismogram analysis and earthquake detection,” *Rev. Geophys.*, vol. 60, 2022.
[10] H. Kanamori, “Seismological observations and theories: From the seismograph to earthquake prediction,” *Annu. Rev. Earth Planet. Sci.*, vol. 49, pp. 61–89.
[11] C. M. Rosca and A. Stancu, “Earthquake Prediction and Alert System Using IoT Infrastructure and Cloud-Based Environmental Data Analysis,” *Appl. Sci. 2024, Vol. 14, Page 10169*, vol. 14, no. 22, p. 10169, Nov. 2024, doi: 10.3390/APP142210169.
[12] Z. T. AlAli and S. A. Alabady, “A survey of disaster management and SAR operations using sensors and supporting techniques,” *Int. J. Disaster Risk Reduct.*, vol. 82, p. 103295, Nov. 2022, doi: 10.1016/J.IJDRR.2022.103295.
[13] “Gated Recurrent Neural Networks Empirical Utilization for Time Series Classification | IEEE Conference Publication | IEEE Xplore.” Accessed: Jun. 17, 2025. [Online]. Available: https://ieeexplore.ieee.org/document/8875282
[14] H. Han *et al.*, “Advanced series decomposition with a gated recurrent unit and graph convolutional neural network for non-stationary data patterns,” *J. Cloud Comput.*, vol. 13, no. 1, pp. 1–19, Dec. 2024, doi: 10.1186/S13677-023-00560-1/FIGURES/7.
[15] Y. Kong *et al.*, “Unlocking the Power of LSTM for Long Term Time Series Forecasting,” Aug. 2024, Accessed: Jun. 17, 2025. [Online]. Available: https://arxiv.org/pdf/2408.10006
[16] P. Pierleoni, R. Concetti, S. Marzorati, A. Belli, and L. Palma, “Internet of Things for Earthquake Early Warning Systems: A Performance Comparison Between Communication Protocols,” *IEEE Access*, vol. 11, pp. 43183–43194, 2023, doi: 10.1109/ACCESS.2023.3271773.
[17] M. S. Abdalzaher, M. Krichen, D. Yiltas-Kaplan, I. Ben Dhaou, and W. Y. H. Adoni, “Early Detection of Earthquakes Using IoT and Cloud Infrastructure: A Survey,” *Sustain. 2023, Vol. 15, Page 11713*, vol. 15, no. 15, p. 11713, Jul. 2023, doi: 10.3390/SU151511713.
[18] A. Alphonsa and G. Ravi, “Earthquake early warning system by IOT using Wireless sensor networks,” *Proc. 2016 IEEE Int. Conf. Wirel. Commun. Signal Process. Networking, WiSPNET 2016*, pp. 1201–1205, Sep. 2016, doi: 10.1109/WISPNET.2016.7566327.
[19] A. Alsehaimi, M. Houda, A. Waqar, S. Hayat, F. Ahmed Waris, and O. Benjeddou, “Internet of things (IoT) driven structural health monitoring for enhanced seismic resilience: A rigorous functional analysis and implementation framework,” *Results Eng.*, vol. 22, p. 102340, Jun. 2024, doi: 10.1016/J.RINENG.2024.102340.
[20] N. N. V. Gupta, M. S. Rinesh, and U. G. Scholar, “IOT BASED EARTHQUAKE DETECTION BY THINGSPEAK”, Accessed: Jun. 17, 2025. [Online]. Available: http://www.acadpubl.eu/hub/
[21] F.e Khan, “Machine learning for earthquake prediction: Applications of lstm and cnns,” *Earthq. Sci. J.*, vol. 17, no. 2, pp. 211–224.
[22] J. Jia and W. Ye, “Deep Learning for Earthquake Disaster Assessment: Objects, Data, Models, Stages, Challenges, and Opportunities,” *Remote Sens.*, vol. 15, no. 16, Aug. 2023, doi: 10.3390/RS15164098/S1<SPAN.
[23] R.e Dutta, “Seismic data analysis for earthquake prediction in the asia-himalaya region,” *J. Seismol. Earthq. Eng.*, pp. 301–315, 2021.
[24] A.e Rashid, “Gru models for efficient earthquake prediction in time-series data,” *Earth Sp. Sci.*, vol. 7, no. 10, pp. 515–530.

[25] Ilham Muthahhari and Muhammad Dzakwan Firdaus, “IoT-Based Seismic Sensor Network Design for Early Warning System in Kalimantan : Literature Review,” *J. Comput. Phys. Earth Sci.*, vol. 4, no. 2, Mar. 2025, doi: 10.63581/JOCPES.V4I2.02.
[26] A.e Kaur, “Cloud computing for real-time seismic data processing: Addressing low-latency requirements,” *J. Cloud Comput. Adv.*, vol. 14, no. 2, pp. 135–148.
[27] J.K. Lee, “Real-time seismic data analysis using iot technology,” *J. Geophys. Eng.*, vol. 72, no. 4, pp. 865–874.
[28] R. Bhasin *et al.*, “Landslide hazards and mitigation measures at Gangtok, Sikkim Himalaya,” *Eng. Geol.*, vol. 64, no. 4, pp. 351–368, Jun. 2002, doi: 10.1016/S0013-7952(01)00096-5.
[29] A.N. Chakraborty, “Advances in seismic event classification us- ing machine learning: A review,” *J. Seismol.*, 2022.
[30] J.P. Kim, “Multi-feature fusion networks for seismic signal classification,” *Geophys. J. Int.*.
[31] Z.H. Zhang, “Time-series feature engineering for seismic magnitude prediction with deep learning,” *Geosci. Front.*.
[32] S. M. Mousavi, W. L. Ellsworth, W. Zhu, L. Y. Chuang, and G. C. Beroza, “Earthquake transformer—an attentive deep-learning model for simultaneous earthquake detection and phase picking,” *Nat. Commun.*, vol. 11, no. 1, Dec. 2020, doi: 10.1038/S41467-020-17591-W,.
[33] M. Nakano, “Efficient preprocessing techniques for seismic data using 1d cnns,” *J. Geophys. Res. Solid Earth.*
[34] V. Koushik, N. V. S. Sasipreetham, M. Nithya, and P. V. Manitha, “Early Detection and Warning System for Earthquakes Using Internet of Things,” *Proc. - 2024 3rd Int. Conf. Sentim. Anal. Deep Learn. ICSADL 2024*, pp. 529–533, 2024, doi: 10.1109/ICSADL61749.2024.00092.
[35] “Magnitude and intensity: Measures of earthquake size and severity,” *Earthq. Inf. Bull.*, vol. 14, no. 6, pp. 209–219, 1982.
[36] X.e Yuan, “Enhancements in gru-based earthquake prediction models using bidirectional layers,” *Seismol. Res. Lett.*, vol. 95, no. 4, pp. 982–990.
[37] M. A. Bilal, Y. Ji, Y. Wang, M. P. Akhter, and M. Yaqub, “Early Earthquake Detection Using Batch Normalization Graph Convolutional Neural Network (BNGCNN),” *Appl. Sci. 2022, Vol. 12, Page 7548*, vol. 12, no. 15, p. 7548, Jul. 2022, doi: 10.3390/APP12157548.