Earthquake Forecasting Using CNN-BiLSTM: Integrating Fuzzy Logic and Reinforcement Learning for Uncertainty Optimization

Mohammed Abduljaleel Shaneen and Suhad Mallalah Kadhem
Department of Computer Science, University of Technology, 10066 Baghdad, Iraq
cs.22.16@grad.uotechnology.edu.iq, suhad.m.kadhem@uotechnology.edu.iq

Keywords: Bidirectional Long Short-Term Memory Networks, Convolution Neural Network, Reinforcement Learning, Fuzzy System.

Abstract: Due mostly to the inherent uncertainty in seismic events, earthquakes prediction is still one of the difficult. A novel hybrid approach is presented to solve this: it is a combination of intelligent rule-based systems and deep learning (DL) architectures. This model uses Bidirectional Long Short-Term Memory (BiLSTM) for temporal sequence modeling, and Convolutional Neural Networks (CNN), for spatial pattern recognition. A fuzzy inference system (FIS) helps managing prediction's uncertainty. Using a agent based on Q-learning, the fuzzy rule base is tuned dynamically for the purpose of increasing performance over time. The model has been trained on 11,442-earthquake event dataset. Attaching a Mean Absolute Error (MAE) of 0.014, minimum Root Mean Squared Error (RMSE) of 0.019, and a R2 score of 0.89 explaining 89% of the data variance, experimental findings show performance gains. These findings demonstrate the excellent ability regarding the proposed framework for controlling uncertainty, hence offering valuable information with regard to proactive risk reduction.

1 INTRODUCTION

Earthquakes, which are among the most powerful natural disasters, can occur without warning and wreak significant destruction, including significant loss of life and significant economic damage to the affected communities [1]. However, they have the potential to trigger a series of secondary disasters, such as tsunamis [2], floods [3], avalanches, and structural collapses brought on through falling debris [4]. Soil liquefaction [5] and fault line rupturing [6] are two examples of such events that can have a major impact on the environment.

Due to the extensive damage caused by seismic events, their high death rates, and their direct and indirect effects, the scientific community has long sought to increase public knowledge and understanding of these phenomena [7], [8], and [9]. Numerous earthquake prediction techniques have been studied over time [10], emphasizing the importance of timely and precise predictions for supporting increased preparedness and risk mitigation initiatives. Reduction of the impact of disasters majorly rests on the prediction of crises events with precision and well-organized responses.

Through increasing preparedness and guaranteeing faster reaction times throughout crises, accurate forecasts help communities to be resilient. Within a certain temporal frame, an ideal forecasting model might be able to estimate geographical location and magnitude of an approaching earthquake [11], therefore perhaps minimizing economic disturbance and averting loss of life.

While several forecast techniques were developed considering different seismic criteria, very few have consistently produced accurate findings. This limitation mostly results from the multifaceted and unpredictable nature of earthquakes, which entail a broad spectrum of complicated characteristics difficult to adequately analyze and model [12].

Though progress in the field has made it possible, precisely and reliably forecasts for earthquake location and timing remain a key scientific difficulty mostly owing to the intrinsic uncertainty that hinder effective prediction. Using BiLSTM and CNN networks, augmented with an Attention Mechanism (AM) and a Fuzzy System, the major goal of the present study is to increase prediction accuracy and solve inherent uncertainty.

Moreover, the Fuzzy System is continuously improved using RL, which helps it to adaptively refine decision boundaries depending on feedback

from prediction performance, hence strengthening the precision and robustness of the model. The study is divided into relevant studies, technique comprising preliminaries, data preparation, the suggested approach, evaluation and results, and conclusion.

2 RELATED WORKS

Notable methods in earthquake prediction research include fuzzy systems, RL systems, shallow ML, and DL algorithms.

Techniques of ML: In essence, those techniques are non-parametric, data-centric, and typically operate with minimal presumptions.

Murwantara et al. [13], the presented work projected the depth, location and magnitude regarding the earthquakes over Indonesia using neural networks (NNs), such as support vector machines (SVM). According to their investigation, SVM algorithm achieved greater predicted accuracy across the different models. Support vector regression and hybrid NN models were also combined by U. Khalil [14] for enhancing earthquake prediction at the Chaman fault in Baluchistan.

Likewise, W. Lin [15], two back-propagation NN models were utilized in the research for detecting significant earthquake events in Taiwan. However, ML approaches might not perform as well in the case when simulating the nonlinear and complex dynamics included in earthquake data; therefore, they typically rely on advanced feature engineering to enhance performance. Huang et al. [16] created a CNN-based model for Taiwanese earthquake magnitudes.

DL algorithms. Recent advances in DL have been quite helpful in overcoming some of the challenges associated with earthquake prediction. CNNs and LSTM networks [17] are two examples of models that have utilized DL architectures.

Al Banna et al. [18] meanwhile suggested an advanced architecture for predicting seismic activity in Bangladesh by use of attention-based bidirectional LSTM networks. Though they still need high-quality datasets for and large computational resources best performance, such models show better capacity in identifying spatial and temporal dependencies within seismic data.

FIS. Grounded in spatial analysis of magnitude distribution, R. Kamat and R. Kamath [19] presented the automated clustering-based adaptive neuro-fuzzy inference system (ANFIS). Though they are flexible, these systems may find it challenging to extract concise, generalizable rules for practical earthquake preparedness.

V. Okumus and A. Mangir [20] also developed a fuzzy logic-based assessment method for quick hazard evaluation regarding reinforced concrete structures under seismic stress, therefore allowing expert knowledge to be used in risk level classification. Pre-earthquake evaluations benefit especially from this model since it gives authorities practical information for preventative safety precautions.

For the prediction of earthquake-induced building damage and necessary rehabilitation [21], present an Adaptive Fuzzy C-Means-based Support Vector Machine (AFCM-SVM) model. The model obtained a high F1-score of 0.945 by means of xBD dataset, feature extraction, and image segmentation, therefore surpassing conventional CNN as well as Siamese networks.

Through connecting geodynamic and seismic processes [22], addresses methods for estimating significant earthquakes. It suggests adding the concept of "information certainty" to raise prediction accuracy and utilizing energy of seismic activity.

Systems for RL [23]: offers a deep RL method for the inversion of earthquake focal mechanisms, therefore enabling self-learning free from extensive training data. Applied to the Mw 7.1 Ridgecrest earthquake, the approach has promise to improve the knowledge of marine earthquake and tsunami evaluation.

EQBot is a deep RL learning-based system for entirely automated earthquake location introduced by [24]. Equipped with ten years of seismic data, EQBot marks a major advancement towards automated seismic processes by aligning S and P waves to precisely identify earthquake epicenters with low error. Notwithstanding such developments, various research ignores the simultaneous collecting of temporal and spatial data, which is necessary for correct predictions.

Review of studies reveals the necessity of more efficient techniques that can simultaneously extract spatial and temporal features while reducing prediction errors. In this setting, a unique hybrid CNN-BiLSTM technique that depends on an AM and FIS for predicting both the location of earthquakes and their quantities within a given time period in Mainland China. This method combines CNN BiLSTM's strengths with the AM dynamically evaluating each feature's significance in the input data to improve the uncertainty depending on FIS.

3 DATA PREPARATION

This section contains the first and second data collection and preprocessing.

3.1 Data Collection

This work made use of a spatial-temporal earthquake dataset derived from the official websites of the National Seismological Center (NSC) and US Geological Survey (USGS). From January 15, 1966, until May 22, 2021, the 11,442 documented events with magnitudes of 3.5 or higher make up the earthquake catalog. Seismic data from the NSC and USGS were used for this study due to their high reliability and accuracy in recording seismic events and their global reliance on researchers.

Every entry in the dataset offers important information including longitude, latitude, occurrence time, depth, magnitude, and station number among other things. Two separate case studies are taken under examination in this work. Whereas Case Study 2 concentrates on maximum recorded magnitude per month, Case Study 1 uses output and input to show the monthly count regarding earthquake occurrences. Combined over monthly intervals, the dataset includes 665 data points. The dataset has been normalized by applying min-max scaling to bring values within a [0, 1] range. For the development of the model, 80% of data has been set aside for training, while the remaining 20% was set aside for testing requirements.

Each seismic event has a spatial image, which is used to extract spatial features via a CNN and temporal features via a BiLSTM network to represent the interconnectedness of seismic events. A monthly time window was used to balance short- and long-term patterns, with min-max normalization and data imbalance addressed by adjusting sample weights during training to improve model performance and accuracy.

3.2 Preprocessing

Throughout the preparation of spatial and temporal data for the prediction of earthquakes using DL, various methods of pre-processing have been employed for the maximization of model performance. Using methods of image enhancement, like noise reduction and histogram equalization (HE), the spatial data's visual quality was improved. Through efficiency, more consistent training was made possible through normalized pixel intensities and standardized image dimensions of 512×512 pixels.

Model generalization capacity as well as dataset variability were further increased through using techniques of data augmentation like geometric transformations and color jittering. Reliability was ensured by temporal data integrity preservation with cleaning methods including outlier removal and interpolation for the purpose of managing missing entries. Retrieval of important time-based features has been performed, and lag features have been added for the purpose of recording sequential dependencies in data.

Efficient model learning was facilitated by scaling and normalizing, either by Min-Max normalization or standardization. Lastly, sequential input samples were created using a sliding window approach, and spatial and temporal features were integrated using feature concatenation. For increasing model robustness and ensure optimal performance, cross-validation as well as hyperparameter tuning were used throughout the training process.

4 METHODOLOGIES

This section contains the details of preliminaries and proposed system.

4.1 Preliminaries

This subsection illustrates the details of Convolutional Neural Network, Bidirectional Long Short-Term Memory, and Attention Mechanism.

4.1.1 Convolutional Neural Network (CNN)

CNNs are widely utilized for processing spatial data, such as images associated with geographic features or earthquake events. CNN architecture is often made up of numerous fundamental parts that work together for extracting and processing spatial patterns [25].

1) Convolution Layer. This layer enables the network for automatically locating as well as extracting local features by applying a set of learnable filters to input matrix. The convolution operation is expressed mathematically as:

$$Y[i,j] = \sum_{m=-k}^{k} \sum_{n=-k}^{k} X[i+m, j+n] \cdot \cdot W[m+k, n+k]. \quad (1)$$

W denotes filter weight; X represents the input matrix; and Y denotes the resulting feature map.

Through utilizing weight sharing as well as local receptive fields, the convolutional technique significantly lowers the number of trainable parameters, enhancing the model's ability for learning spatial feature hierarchies.

2) Pooling Layer. Pooling layers assist in down-sampling the feature maps following the convolution layer, reducing their dimensionality and avoiding overfitting. A popular pooling operation, like max pooling, is defined in the following way:

$$Y' = \max_{(i,j)\in R} Y[i,j], \quad (2)$$

where Y'denotes the output of pooling operation; R denotes the pooling region.

3) Fully Connected (FC) Layers. Following the final pooling process, the produced feature maps are 1D vectors passed through one or more FCLs. Learning complex nonlinear relations among the obtained features falls on such layers. One can express the transformation as:

$$Z = W \cdot A + b, \quad (3)$$

where W denotes weight matrix, A denotes the flattened output, and b denotes bias [26].

4.1.2 Bidirectional Long Short-Term Memory (BiLSTM)

The techniques of DL prioritize extracting high-level aspects, through the differentiation from other kinds of approaches [27]. CNNs and BiLSTM are considered as a sub-set of DL methods [28], where the BiLSTM networks are quite efficient at capturing the temporal dependencies in the sequences of earthquakes, as it exhibits high scalability and low complexity [29].

Particularly suitable for evaluating earthquake-related time series including depth, magnitude, and timestamp information, the BiLSTM network is a specialized architecture designed for processing sequential data. Through efficient learning of long-range dependencies and preservation of contextual relations across sequences, the model surpasses the limitations of conventional Recurrent Neural Networks (RNNs).

1) LSTM Structure. Three basic gates regulate the information flow in each individual LSTM cell [30]:

Input Gate. Determines the degree of contribution the incoming input makes to the cell state. It comes out as:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i). \quad (4)$$

Forget Gate. Controls the removal of irrelevant data from previous cell state:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f). \quad (5)$$

Output Gate. Determines whether parts of the cell must be output to the next hidden state:

$$o_t = \sigma(W_o \cdot [h_{t-1}, C_t] + b_o). \quad (6)$$

2) Updating Cell State. LSTM cell state serves the network as its memory component. The values regarding the new candidate as well as the memory that was retained from the previous step are merged to update it: The candidate cell state is first calculated in the following way:

$$\hat{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c). \quad (7)$$

After that, the new cell state is updated with the use of:

$$C_t = f_t \odot C_{t-1} + i_t \odot \hat{C}_t. \quad (8)$$

Here, $\odot$ denotes the element-wise multiplication, which allows the model for selectively retaining or incorporating new information.

3) Hidden State Output. Hidden state output utilized for the final output or the next time step, is produced by the filtration of updated cell state over the output gate:

$$h_t = o_t \odot \tanh(C_t). \quad (9)$$

4) BiLSTM Structure. BiLSTM consists of two parallel LSTM layers, one of them moves forward through sequence and the other one backwards. The model gains a comprehensive understanding of temporal context from past as well as future states thanks to this bidirectional technique. At time t, the combined hidden state is interpreted as follows:

$$h_t = h_t^{\text{forward}} \oplus h_t^{\text{backward}}. \quad (10)$$

In this equation, $\oplus$ represents summation or concatenation of hidden states from 2 directions.

Extremely helpful in situations of time series analysis, BiLSTM model could be effective in capturing complex temporal patterns by the use of dual-sequence processing and gated processes.

4.1.3 Attention Mechanism (AM)

AM plays a key role to improve model accuracy by allowing it to focus dynamically on the most pertinent input sections that contribute substantially to

prediction task [31].

1) Attention Score Calculation. Attention score is estimated as follows [32]:

$$s_t = \tanh(W_h h_t + b_h), \quad (11)$$

where s_t represent the attention score for time step, W_h represent a learnable weight matrix, and b_h represent the bias.

2) Normalization with Softmax. The raw attention scores can be converted into a probability distribution with the aid of softmax function. This ensures that the scores are normalized over all time steps:

$$a_t = \frac{e^{s_t}}{\sum_j e^{s_j}}. \quad (12)$$

The resulting a_t values denote the relative importance regarding every hidden state in the overall sequence context.

3) Final Attention Value. Combining the attention scores with the corresponding hidden states results in the attention-weighted representation:

$$s = a_t h_t. \quad (13)$$

This output vector efficiently emphasizes the most important aspects, therefore allowing the model to produce more contextually informed predictions [33].

4.2 Proposed System

The suggested system is shown in Figure 1.

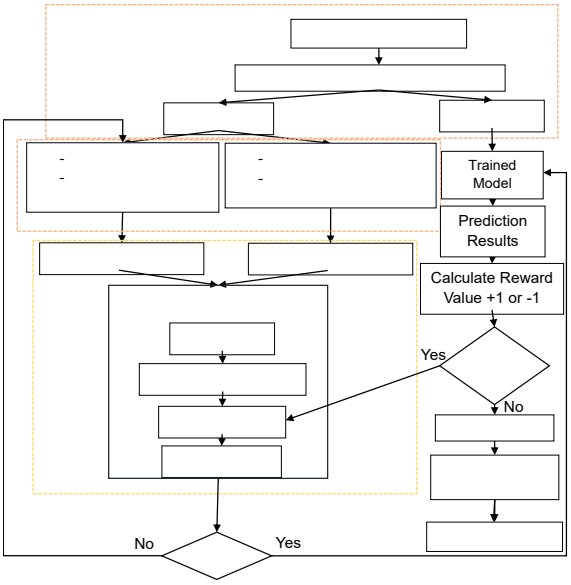

Figure 1: Proposed System Flowcha.

4.2.1 Framework Architecture and Methodological Workflow

Combining CNN, BiLSTM, and an AM, the suggested earthquake prediction model presents a unified predictive framework. Five sequential stages, each corresponding to a different phase in the processing pipeline, comprise this architecture.

4.2.1.1 Data Preparation Block

The initial phase consists of organizing and preprocessing the data required to train a model. First gathered, after that normalized to provide a consistent numerical range across features an important step for steady and effective training are earthquake records. The data-set is then separated to a testing set for performance assessment as well as a training set utilized for model learning.

Furthermore, used throughout preprocessing to maintain temporal consistency in the data is Zero-Order Hold (ZOH) method, therefore enabling the model to better understand temporal patterns and trends that are inherent in seismic activities.

4.2.1.2 Feature Extraction Block

As described in Table 1, this block consists of five convolutional layers each matched with a max-pooling layer. These layers extract spatial properties from the input data in turn. While the pooling layers lower spatial resolution and hence control overfitting and computational load, the convolutional procedures capture local patterns. Subsequent the final pooling operation, a flattening layer converts the multidimensional feature maps to a 1D vector, therefore preparing data for subsequent sequence-learning stage.

Table 1: CNN Details.

Layer Type	Details
Convolutional	Conv1: 16 filters, size 5, stride 1
Pooling Layer	Max-Pooling1: Pool size 2, stride 1
Convolutional	Conv2: 32 filters, size 3, stride 1
Pooling Layer	Max-Pooling2: Pool size 2, stride 1
Convolutional	Conv3: 64 filters, size 3, stride 1
Pooling Layer	Max-Pooling3: Pool size 2, stride 1
Convolutional	Conv4: 128 filters, size 3, stride 1
Pooling Layer	Max-Pooling4: Pool size 2, stride 1
Convolutional	Conv5: 256 filters, size 3, stride 1
Pooling Layer	Max-Pooling5: Pool size 2, stride 1
Flatten Layer	Converts multi-dimensional output to 1D vector for further processing

4.2.1.3 Sequence Learning Block

Using two BiLSTM layers, this stage learns from the temporal nature of earthquake sequences. There are 128 units in the first layer; the second includes 64 units; dropout layers help to reduce overfitting.

BiLSTM's internal mechanics consist in gate-based processes. The Input Gate chooses which of the current inputs are relevant as in (4). The Forget Gate controls which data from previous cell state must be discarded as in (5). The output gate controls the hidden state contribution regarding the cell as in (6). controls the cell state's output. The cell state Ct is updated through (8). The hidden state is generated using (9). Through such processes, BiLSTM is able to acquire the complex temporal patterns.

4.2.1.4 Attention Mechanism Step

To allow the model to highlight the most relevant temporal characteristics, AM is added after the CNN and BiLSTM layers. This mechanism is defined by three fundamental steps. Initially, the equations aids in calculating the attention score as in (11). This score quantifies the significance regarding the hidden state ht by converting it through a learned weight matrix Wh as well as bias bh. The equation after that shows how the Softmax function normalizes the scores as in (12). The model can determine the relative value regarding each feature in the input sequence framework thanks to this normalization phase, which ensures that the attention scores are converted into probabilities that add up to one.

The computed final attention value is as (13) In this stage, the normalized attention weights are combined with associated hidden states for creating a feature representation that emphasizes the most significant features of the input.

4.2.1.5 Prediction Block

The proposed model's final component is termed Prediction Block, consisting of a series of FC layers that have been designed for providing the final output. This block includes 3 successive layers: layer 1 (FC1) has 32 neurons, layer 2 (FC2) contains 10, and layer 3 (FC3), the output node, contains 1 neuron.

Through converting high-level representations taken from spatial and temporal modules to accurate capture of complex feature interactions, this architectural design aids the model to produce accurately predicted outputs. After completing the training, the model's performance is evaluated thoroughly with the use of a specialized testing data-set to confirm that it is capable.

4.2.2 Integration with Fuzzy Inference System (FIS)

Outputs of CNN and BiLSTM module are processed further by an FIS for the enhancement of the interpretability in addition to lowering uncertainty of predictions. The pipeline for integration is structured:

1) Fuzzification. Numerical outputs of CNN and BiLSTM models are converted to linguistic variables like "low risk," "medium risk," and "high risk.".
2) Membership Functions. The degree to which a crisp input belongs to a fuzzy set is determined by membership functions:

$$\mu_{\text{low}}(x) = \max\left(0, \min\left(1, \frac{c-x}{c-a}\right)\right), \quad (14)$$

$$\mu_{\text{high}}(x) = \max\left(0, \min\left(1, \frac{x-b}{c-b}\right)\right). \quad (15)$$

The degree of risk associated with each input value is measured by such functions.

3) Inference Rules. The rule base is constructed using empirical patterns as well as expert knowledge.
   - If B is "high risk," then C is "high risk," then the prediction is "high likelihood of earthquake." Sample fuzzy rules.
   - In the event that B (BiLSTM result) and C (CNN result) are both "high risk," the prediction is "high likelihood of earthquake."
   - The prediction is "high likelihood of earthquake," if C (CNN result) is "high risk," and depth D is less than 10km.
   - The prediction is "medium likelihood of earthquake," should B (BiLSTM result) be "medium," A (magnitude) be "high," and C (CNN result) be "high risk."
   - The prediction is "high likelihood of earthquake," should the time interval T between previous earthquakes be less than five years and C (CNN result) be "high risk."
   - If D is between 10 km and 30 km and B is "medium risk", then the prediction is "medium probability of earthquake."
   - If A is "low" and C is "high risk," then the prediction is "medium earthquake."
   - If B is "high risk and T is less than two years" then the prediction result is "high probability of earthquake."
4) Defuzzification. The centroid approach converts the fuzzy result of applying the rule into a precise prediction:

$$Z = \frac{\sum_i (z_i \cdot \mu(z_i))}{\sum_i \mu(z_i)}. \quad (16)$$

This allows to generate interpretable, scalar predictions while preserving robustness in uncertain contexts through including BiLSTM and CNN outputs into the FIS.

4.2.3 Integration of Reinforcement Learning for Fuzzy Rule Optimization

To further increase the FIS's performance, the proposed framework incorporates an RL module. By interacting with the environment, the module primarily seeks to dynamically optimize the fuzzy inference rules based on the prediction outcomes.

4.2.3.1 Objective and Role

RL agent is intended to function as an autonomous controller that assesses and modifies the fuzzy rule base. With the use of feedback signals derived from the accuracy of predictions, it iteratively refines the inference rules. Proper predictions provide positive results, whereas incorrect predictions have negative results. The system learns the optimal policy for managing the rule base to maximize prediction performance.

4.2.3.2 Reinforcement Learning Formulation

Formally, RL process is modeled as a Markov Decision Process (MDP), in this case:

- State (S) denotes the input-output mapping state as well as the current configuration of fuzzy rules.
- Action (A) denotes to possible modifications to the fuzzy rule base include changing rule weights, adding/removing rules, or membership thresholds.
- Reward (R): Depending on the prediction accuracy of the model, a scalar feedback signal assigned +1 for correct predictions and −1 for incorrect predictions.
- Policy (π) is a method assigning the most suitable action to each one of the states.

4.2.3.3 Learning Algorithm

With the use of the update rule [34], a value-based algorithm—more especially, Q-Learning—updates the predicted utility of actions.

$$\alpha\left[r+\gamma\max_{'a}Q(s',a')-Q(s,a)\right]+Q(s,a)\leftarrow Q(s,a). \quad (17)$$

Where Q(s,a) is the estimated value of taking action a in states, $\alpha\in(0,1)$ represents learning rate, $\gamma\in(0,1)$ is the discount factor for future rewards, s′ is the new state reached after action a is taken.

4.2.3.4 Rule Base Adaptation Strategy

The agent constantly updates fuzzy rule base depending on feedback. This technique includes:

- Reinforcing highly predictive rules through raising their confidence weights.
- Pruning or penalizing rules that support erroneous predictions.
- Investigating the rule space is helpful in finding new rule combinations enhancing accuracy.

4.2.3.5 Expected Outcomes

By adding RL, FIS becomes adaptable and self-correcting, moving from static rule-based model to dynamic system that can adapt to new patterns.

5 RESULTS AND EVALUATION

This section discusses results and evaluation of the proposed system.

5.1 Number of Earthquakes

Evaluated over nine different seismic regions. The comparison models were chosen to include CNN, LSTM, and CNN-BiLSTM for their direct connection to the proposed deep learning architecture, as well as SVM, DT, MLP, and RF for having been previously used on the same earthquake dataset. Three generally utilized evaluation measures are utilized like RMSE, MAE, and R^2 to assess the prediction performance of any model, as described in Table 2 and based on [35]:

$$\mathrm{RMSE}=\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}, \quad (18)$$

$$\mathrm{MAE}=\frac{1}{n}\sum_{i=1}^{n}|y_i-\hat{y}_i|, \quad (19)$$

$$R^2=1-\frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{\sum_{i=1}^{n}(y_i-\bar{y})^2}. \quad (20)$$

The experimental findings demonstrate that the proposed method for earthquake prediction significantly outperforms more conventional models. The system in region 1 achieved an RMSE of 0.019, which is a 20.83% reduction when compared to BiLSTM model, which generated an RMSE of 0.024. The proposed framework revealed a value much higher than RF model, which produced a 0.044 MAE, with a 0.014 MAE. The model had an explanatory power value of 0.890, which means that 89%

regarding the variability in the observed data could be adequately described.

Figure 2 illustrates how much better the performance is over models such as LSTM, which achieved a rather low R2 of 0.174. Those findings demonstrate the effectiveness of integrating CNN with BiLSTM as well as FIS, which combined improve the model's capacity to identify complex seismic patterns.

5.2 Maximum Magnitude of Earthquake

In order to reduce possible human and financial losses, this case addresses the prediction of earthquake epicenter as well as maximum magnitude. When compared to traditional approaches, Table 3 demonstrates the proposed model's superior ability to capture seismic patterns and increase forecast accuracy.

Table 2: Performance comparison of models across different regions.

Region	Metric	SVM	DT	MLP	RF	CNN	LSTM	CNN-BiLSTM	Proposed Method
1	RMSE	0.1030	0.0940	0.1070	0.0940	0.0950	0.0930	0.0240	0.0190
	MAE	0.0610	0.1080	0.0480	0.0440	0.0510	0.0490	0.0180	0.0140
	R^2	-0.011	0.1570	0.0500	0.1540	0.1430	0.1740	0.7560	0.8900
2	RMSE	0.1250	0.0940	0.1450	0.1280	0.0830	0.0730	0.0640	0.0580
	MAE	0.0860	0.1080	0.0920	0.0580	0.0490	0.0420	0.0310	0.0270
	R^2	0.6040	0.6600	0.4320	0.5840	0.8260	0.8640	0.7150	0.8320
3	RMSE	0.2820	0.2590	0.1820	0.2010	0.1990	0.1920	0.0690	0.0620
	MAE	0.1520	0.1520	0.1330	0.1120	0.1120	0.1190	0.0300	0.0260
	R^2	-0.226	-0.030	0.4340	0.3700	0.4120	0.4500	0.7370	0.8610
4	RMSE	0.1960	0.1230	0.0970	0.1060	0.0920	0.0760	0.0590	0.0480
	MAE	0.0880	0.0840	0.0720	0.0770	0.0690	0.0650	0.0620	0.0510
	R^2	0.0110	0.1380	0.2630	0.2140	0.4320	0.4950	0.6970	0.7450
5	RMSE	0.2290	0.1970	0.1740	0.1920	0.1950	0.1960	0.0880	0.0680
	MAE	0.1250	0.0820	0.0710	0.0750	0.0720	0.0700	0.0470	0.0430
	R^2	-0.049	0.5830	0.7490	0.6840	0.6950	0.7360	0.7050	0.8860
6	RMSE	0.1590	0.1640	0.1060	0.0950	0.0930	0.0930	0.0990	0.0770
	MAE	0.1420	0.1250	0.0110	0.1260	0.1400	0.1320	0.0600	0.0530
	R^2	-0.347	0.0190	0.2130	0.0590	0.0540	0.0960	0.8050	0.8290
7	RMSE	0.1410	0.1410	0.1340	0.1410	0.1300	0.1310	0.0920	0.0830
	MAE	0.1020	0.1380	0.0490	0.0390	0.0330	0.0350	0.0260	0.0230
	R^2	0.1320	0.1100	0.2190	0.3650	0.4310	0.4580	0.6090	0.6450
8	RMSE	0.1220	0.1220	0.1240	0.1190	0.1200	0.1150	0.0870	0.0770
	MAE	0.0780	0.0740	0.0670	0.0750	0.0580	0.0560	0.0590	0.0480
	R^2	0.0750	0.1820	0.2640	0.1800	0.3920	0.3850	0.8120	0.8420
9	RMSE	0.6120	0.5090	0.4750	0.5240	0.5600	0.5730	0.8120	0.7900
	MAE	0.0780	0.0910	0.0990	0.0930	0.0910	0.0870	0.0590	0.0480
	R^2	0.6120	0.5090	0.4750	0.5240	0.5600	0.5730	0.8120	0.8420

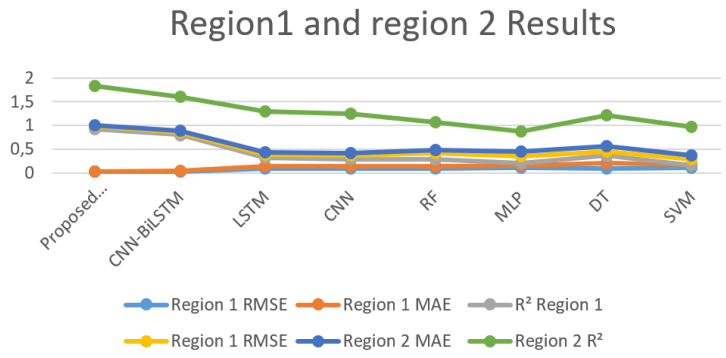

Figure 2: Performance comparison of the proposed system vs. traditional models.

The evaluation demonstrates that the recommended approach performs better than conventional ML models in earthquake prediction. It consistently produces lower MAE and MAE across all sectors, indicating increased precision and reliability.

The model's ability to incorporate data variance is further demonstrated by the significantly higher R^2 values. When combined, CNNs, BiLSTMs, AMs, and fuzzy logic improve the model's ability to comprehend complex seismic patterns. Overall, the results demonstrate the system's strength and potential to improve earthquake prediction as well as disaster risk reduction. A comparison of areas 1 and 2 is presented in Figure 3.

Table 3: Performance comparison for predicting location and maximum earthquake magnitude across different regions.

Region	Metric	SVM	DT	MLP	RF	CNN	LSTM	CNN-BiLSTM	Proposed Approach
1	RMSE	0.4410	0.4370	0.1040	0.4420	0.2640	0.2400	0.0210	0.0130
	MAE	0.3800	0.2900	0.0400	0.3770	0.0620	0.0540	0.0210	0.0090
	R^2	-0.344	-0.141	0.0520	-0.256	0.0960	0.1640	0.7950	0.9100
2	RMSE	0.4380	0.2340	0.8210	0.1940	0.1280	0.1090	0.0420	0.0370
	MAE	0.2100	0.2070	0.1910	0.1680	0.1750	0.1490	0.0400	0.0230
	R^2	0.5020	0.7220	0.1070	0.6540	0.5920	0.6800	0.7910	0.9020
3	RMSE	0.4380	0.3220	0.3300	0.3400	0.3320	0.3180	0.0700	0.0580
	MAE	0.2840	0.3020	0.2770	0.2840	0.2840	0.2450	0.0620	0.0450
	R^2	-0.251	0.2120	0.2960	0.2460	0.3600	0.4510	0.7620	0.8790
4	RMSE	0.5180	0.4460	0.4280	0.4160	0.4050	0.3700	0.1320	0.1100
	MAE	0.3110	0.2480	0.2620	0.2550	0.2120	0.2250	0.1180	0.0950
	R^2	-0.172	0.0880	0.1560	0.0720	0.2560	0.2610	0.8660	0.8890
5	RMSE	0.1860	0.1650	0.2030	0.1510	0.1490	0.1490	0.0290	0.0220
	MAE	0.1610	0.1430	0.1850	0.1340	0.1360	0.1340	0.0390	0.0280
	R^2	0.2520	0.4240	0.1580	0.5020	0.4970	0.5090	0.7710	0.8900
6	RMSE	0.3980	0.3460	0.3940	0.3450	0.3420	0.3330	0.1200	0.0900
	MAE	0.3580	0.2880	0.3620	0.2850	0.2860	0.2750	0.2720	0.2400
	R^2	-0.301	0.2190	-0.271	0.2430	0.2500	0.3000	0.7210	0.8420
7	RMSE	0.2050	0.1900	0.2090	0.1870	0.1850	0.1840	0.1280	0.1050
	MAE	0.1860	0.1710	0.1980	0.1700	0.1670	0.1670	0.0820	0.0700
	R^2	0.2560	0.4210	0.3020	0.4450	0.5010	0.5080	0.8090	0.8320
8	RMSE	0.1500	0.1470	0.1380	0.1460	0.1170	0.1140	0.0610	0.0500
	MAE	0.0780	0.0720	0.0680	0.0720	0.0600	0.0590	0.0240	0.0180
	R^2	0.1450	0.1780	0.2210	0.1740	0.4210	0.4500	0.7140	0.8450
9	RMSE	0.1620	0.1640	0.1590	0.1630	0.1430	0.1400	0.0720	0.0600
	MAE	0.1350	0.1380	0.1330	0.1370	0.1280	0.1270	0.0570	0.0450
	R^2	0.1420	0.1110	0.1650	0.1240	0.2010	0.2350	0.8010	0.8380

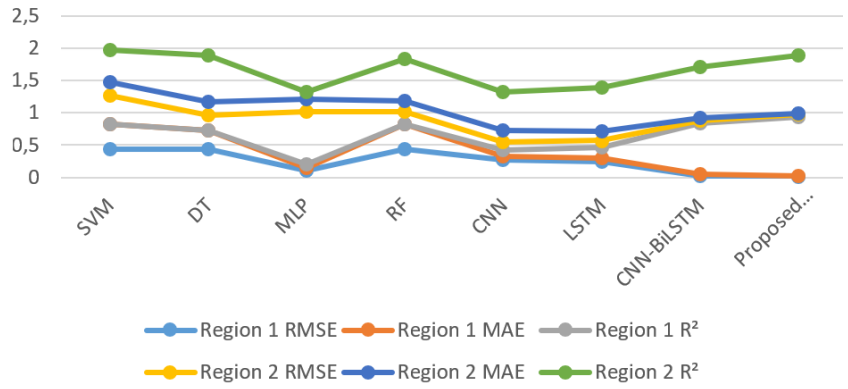

Figure 3: Comparative analysis of models for predicting earthquake maximum magnitude and location in Region 1,2.

6 CONCLUSIONS

For overcoming long-standing challenges with for overcoming long-standing challenges with uncertainty and complexity in seismic forecasting, this research proposes a multi-layered, novel framework for earthquake prediction that combines the power of DL and intelligent systems. By smoothly integrating CNN for spatial feature extraction, an AM for feature prioritization, and BiLSTM networks for temporal pattern learning, the model improves its understanding of earthquake dynamics, and it not only increases predictive accuracy, yet skillfully manages the inherent uncertainty associated with earthquake forecasting by using FIS that enhances interpretability and robustness. whereas integration with reference to RL further permits the framework to evolve adaptively through ongoing optimization of fuzzy rules. The proposed system was tested on a large dataset comprising over 11,000 seismic events, which made it possible to fully train and validate the model. The proposed system notably surpassed traditional models in RMSE, MAE, and R2 for every evaluation metric. The results showed good accuracy using RMSE, MAE, and R^2 metrics, while ensuring the interpretability of the results via fuzzy inference system and embedded reinforcement learning that was used for optimizing inference rules. These findings demonstrate how well the framework can represent nonlinear, high-dimensional relations in seismic data. Beyond its empirical strength, this framework represents a paradigm shift in the use of AI to address geophysical issues since it provides a interpretable and scalable solution for practical implementation in early warning systems. Both disaster resilience and public safety might benefit from more precise and quick risk assessments, which might be made possible by a greater ability to rank key traits and modify decision-making guidelines as needed. Through examining multi-modal sensor integration, adding real-time data streams, and expanding the model's generalizability across numerous geographic regions and apply the model in real time, future research could improve on this work. Ultimately, this architecture lays the foundation for seismic intelligence systems driven by AI that will be able to alter the perceptions of and responses to seismic risks.

REFERENCES

[1] A. S. N. Alarifi, N. S. N. Alarifi, and S. Al-Humidan, "Earthquakes magnitude predication using artificial neural network in northern Red Sea area," Journal of King Saud University - Science, vol. 24, no. 4, pp. 301-313, Oct. 2012, [Online]. Available: https://doi.org/10.1016/j.jksus.2011.05.002.

[2] P. Cui et al., "The Wenchuan Earthquake (May 12, 2008), Sichuan Province, China, and resulting geohazards," Natural Hazards, vol. 56, no. 1, pp. 19-36, Apr. 2009, [Online]. Available: https://doi.org/10.1007/s11069-009-9392-1.

[3] N. Jain, D. Virmani, and A. Abraham, "Proficient 3-class classification model for confident overlap value based fuzzified aquatic information extracted tsunami prediction," Intelligent Decision Technologies, vol. 13, no. 3, pp. 295-303, Sep. 2019, [Online]. Available: https://doi.org/10.3233/idt-180003.

[4] P. Cui, L. Xiang, and Q. Zou, "Risk assessment of highways affected by debris flows in Wenchuan earthquake area," Journal of Mountain Science, vol. 10, no. 2, pp. 173-189, Apr. 2013, [Online]. Available: https://doi.org/10.1007/s11629-013-2575-y.

[5] R. Verdugo and J. González, "Liquefaction-induced ground damages during the 2010 Chile earthquake," Soil Dynamics and Earthquake Engineering, vol. 79, pp. 280-295, Dec. 2015, [Online]. Available: https://doi.org/10.1016/j.soildyn.2015.04.016.

[6] J. D. Bray, "Developing mitigation measures for the hazards associated with earthquake surface fault rupture," University of Tokyo Press, pp. 55-79, 2001.

[7] R. Bilham, "The seismic future of cities," Bulletin of Earthquake Engineering, vol. 7, no. 4, pp. 839-887, Sep. 2009, [Online]. Available: https://doi.org/10.1007/s10518-009-9147-0.

[8] N. N. Ambraseys and C. P. Melville, A History of Persian Earthquakes. Cambridge University Press, 2005.

[9] J. Jia, Modern Earthquake Engineering: Offshore and Land-Based Structures. Springer, 2018.

[10] G. A. Sobolev, "Methodology, results, and problems of forecasting earthquakes," Herald of the Russian Academy of Sciences, vol. 85, no. 2, pp. 107-111, Mar. 2015, [Online]. Available: https://doi.org/10.1134/s1019331615020069.

[11] R. Kail, E. Burnaev, and A. Zaytsev, "Recurrent Convolutional Neural Networks Help to Predict Location of Earthquakes," IEEE Geoscience and Remote Sensing Letters, vol. 19, pp. 1-5, 2022, [Online]. Available: https://doi.org/10.1109/lgrs.2021.3107998.

[12] G. V. Otari and R. V. Kulkarni, "A review of application of data mining in earthquake prediction," International Journal of Computer Science and Information Technologies, vol. 3, no. 2, pp. 3570-3574, 2012.

[13] I. M. Murwantara, P. Yugopuspito, and R. Hermawan, "Comparison of machine learning performance for earthquake prediction in Indonesia using 30 years historical data," TELKOMNIKA (Telecommunication Computing Electronics and Control), vol. 18, no. 3, p. 1331, Jun. 2020, [Online]. Available: https://doi.org/10.12928/telkomnika.v18i3.14756.

[14] U. Khalil, B. Aslam, Z. A. Kazmi, and A. Maqsoom, "Integrated support vector regressor and hybrid neural network techniques for earthquake prediction along Chaman fault, Baluchistan," Arabian Journal of Geosciences, vol. 14, no. 21, Oct. 2021, [Online]. Available: https://doi.org/10.1007/s12517-021-08564-4.

[15] J.-W. Lin, "Researching significant earthquakes in Taiwan using two back-propagation neural network models," Natural Hazards, Jul. 2020, [Online]. Available: https://doi.org/10.1007/s11069-020-04144-z.

[16] J. Huang, X. Wang, Y. Zhao, C. Xin, and H. Xiang, "Large Earthquake Magnitude Prediction in Taiwan Based on Deep Learning Neural Network," Neural Network World, vol. 28, no. 2, pp. 149-160, 2018, [Online]. Available: https://doi.org/10.14311/nnw.2018.28.009.

[17] T. Bhandarkar, V. K, N. Satish, S. Sridhar, R. Sivakumar, and S. Ghosh, "Earthquake trend prediction using long short-term memory RNN," International Journal of Electrical and Computer Engineering, vol. 9, no. 2, pp. 1304-1312, Apr. 2019, [Online]. Available: https://doi.org/10.11591/ijece.v9i2.pp1304-1312.

[18] A. Banna, T. Ghosh, K. A. Taher, M. S. Kaiser, and M. Mahmud, "An Earthquake Prediction System for Bangladesh Using Deep Long Short-Term Memory Architecture," Lecture Notes in Networks and Systems, pp. 465-476, 2021, [Online]. Available: https://doi.org/10.1007/978-981-33-6081-5_41.

[19] R. S. Kamath and R. K. Kamat, "Earthquake magnitude prediction for Andaman-Nicobar Islands: Adaptive neuro fuzzy modeling with fuzzy subtractive clustering approach," Journal of Chemical and Pharmaceutical Sciences, vol. 10, no. 3, pp. 1228-1233, 2017.

[20] A. Mangir and V. Okumus, "Pre-earthquake fuzzy logic-based rapid hazard assessment of reinforced concrete buildings," Case Studies in Construction Materials, vol. 19, p. e02534, Oct. 2023, [Online]. Available: https://doi.org/10.1016/j.cscm.2023.e02534.

[21] S. Pradeep Kumar, L. Jasim, C. H. Basha, S. Rafikiran, and K. Parashu Ramulu, "Predicting Earthquake Damage and Rehabilitation Intervention Using Adaptive Fuzzy C-Means-Based Support Vector Machine," in 2024 International Conference on Integrated Intelligence and Communication Systems (ICIICS), Nov. 2024, pp. 1-6, [Online]. Available: https://doi.org/10.1109/ICIICS63763.2024.108594.

[22] A. V. Solomatin, "On Earthquake Prediction, the Relationship between Seismic and Geodynamic Processes, and the Concept of Information Certainty," Journal of Volcanology and Seismology, vol. 19, no. 1, pp. 67-77, Feb. 2025, [Online]. Available: https://doi.org/10.1134/s0742046324700878.

[23] W. Kuang, Z. Zou, J. Xing, and W. Wei, "Deep reinforcement learning for inverting earthquake focal mechanism and its potential application to marine earthquakes," Intelligent Marine Technology and Systems, vol. 2, no. 1, Jun. 2024, [Online]. Available: https://doi.org/10.1007/s44295-024-00031-6.

[24] W. Kuang, C. Yuan, Z. Zou, J. Zhang, and W. Zhang, "Autonomous Earthquake Location via Deep Reinforcement Learning," Seismological Research Letters, vol. 95, no. 1, pp. 367-377, Oct. 2023, [Online]. Available: https://doi.org/10.1785/0220230118.

[25] Y. Li, Y. He, and M. Zhang, "Prediction of Chinese energy structure based on Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM)," Energy Science & Engineering, 2020, [Online]. Available: https://doi.org/10.1002/ese3.698.

[26] W. Lu, J. Li, Y. Li, A. Sun, and J. Wang, "A CNN-LSTM-Based Model to Forecast Stock Prices," Complexity, vol. 2020, pp. 1-10, Nov. 2020, [Online]. Available: https://doi.org/10.1155/2020/6622927.

[27] A. A. Hussan, S. H. Shaker, and A. E. Ali, "Strangeness Detection from Crowded Video Scenes by Hand-Crafted and Deep Learning Features," Journal of Soft Computing & Computer Applications, vol. 1, no. 1, Jun. 2024, [Online]. Available: https://doi.org/10.70403/3008-1084.1005.

[28] S. F. Abbas, S. H. Shaker, and F. A. Abdullatif, "Face Mask Detection Based on Deep Learning: A Review," Journal of Soft Computing and Computer Applications, vol. 1, no. 1, Art. 1006, 2024, [Online]. Available: https://doi.org/10.70403/3008-1084.1006.

[29] H. K. A. Atheem, I. T. Ali, and F. A. Al Alawy, "A Comprehensive Analysis of Deep Learning and Swarm Intelligence Techniques to Enhance Vehicular Ad-hoc Network Performance," Journal of Soft Computing and Computer Applications, vol. 1, no. 1, Art. 1004, 2024, [Online]. Available: https://doi.org/10.70403/3008-1084.1004.

[30] H. Shi, K. Miao, and X. Ren, "Short-term load forecasting based on CNN-BiLSTM with Bayesian optimization and attention mechanism," Concurrency and Computation: Practice and Experience, 2021, [Online]. Available: https://doi.org/10.1002/cpe.6676.

[31] Y. Chu, X. Yue, L. Yu, M. Sergei, and Z. Wang, "Automatic Image Captioning Based on ResNet50 and LSTM with Soft Attention," Wireless Communications and Mobile Computing, vol. 2020, pp. 1-7, Oct. 2020, [Online]. Available: https://doi.org/10.1155/2020/8909458.

[32] B. Zhang, D. Xiong, J. Xie, and J. Su, "Neural Machine Translation With GRU-Gated Attention Model," IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 11, pp. 4688-4698, Jan. 2020, [Online]. Available: https://doi.org/10.1109/tnnls.2019.2957276.

[33] M. H. A. Banna et al., "Attention-Based Bi-Directional Long-Short Term Memory Network for Earthquake Prediction," IEEE Access, vol. 9, pp. 56589-56603, 2021, [Online]. Available: https://doi.org/10.1109/access.2021.3071400.

[34] A. M. Andrew, "Reinforcement learning: An introduction," Kybernetes, vol. 27, no. 9, pp. 1093-1096, Dec. 1998.

[35] G. James, D. Witten, T. Hastie, and R. Tibshirani, "An introduction to statistical learning: With applications in R," International Statistical Review, vol. 82, no. 1, pp. 156-157, Apr. 2014, [Online]. Available: https://doi.org/10.1111/insr.12051_19.