

Article

Small Earthquakes Can Help Predict Large Earthquakes: A Machine Learning Perspective

Xi Wang ¹, Zeyuan Zhong ¹, Yuechen Yao ¹, Zexu Li ¹, Shiyong Zhou ², Changsheng Jiang ³ and Ke Jia ^{1,4,*}

¹ School of Automation, Northwestern Polytechnical University, Xi'an 710129, China

² School of Earth and Space Science, Peking University, Beijing 100871, China

³ Institute of Geophysics, China Earthquake Administration, No. 5 Minzu Daxue Nan Road, Haidian District, Beijing 100086, China

⁴ Shanghai Sheshan National Geophysical Observatory, Shanghai 201602, China

* Correspondence: jk@nwpu.edu.cn

Abstract: Earthquake prediction is a long-standing problem in seismology that has garnered attention from the scientific community and the public. Despite ongoing efforts to understand the physical mechanisms of earthquake occurrence, there is no convincing physical or statistical model for predicting large earthquakes. Machine learning methods, such as random forest and long short-term memory (LSTM) neural networks, excel at identifying patterns in large-scale databases and offer a potential means to improve earthquake prediction performance. Differing from physical and statistical approaches to earthquake prediction, we explore whether small earthquakes can be used to predict large earthquakes within the framework of machine learning. Specifically, we attempt to answer two questions for a given region: (1) Is there a likelihood of a large earthquake (e.g., $M \geq 6.0$) occurring within the next year? (2) What is the maximum magnitude of an earthquake expected to occur within the next year? Our results show that the random forest method performs best in classifying large earthquake occurrences, while the LSTM method provides a rough estimation of earthquake magnitude. We conclude that small earthquakes contain information relevant to predicting future large earthquakes and that machine learning provides a promising avenue for improving the prediction of earthquake occurrences.



Citation: Wang, X.; Zhong, Z.; Yao, Y.; Li, Z.; Zhou, S.; Jiang, C.; Jia, K. Small Earthquakes Can Help Predict Large Earthquakes: A Machine Learning Perspective. *Appl. Sci.* **2023**, *13*, 6424. <https://doi.org/10.3390/app13116424>

Academic Editor: Roberto Zivieri

Received: 6 April 2023

Revised: 7 May 2023

Accepted: 22 May 2023

Published: 24 May 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The prediction of earthquakes has long been a formidable challenge [1–3], owing to several factors. Firstly, seismic events are the result of intricate interactions between tectonic plates, faults, and other geological factors [4], rendering the accurate forecasting of their timing and magnitudes exceedingly challenging. Secondly, the paucity of long-term and extensive data poses a significant hurdle in earthquake prediction. Large earthquakes often recur at lengthy intervals (hundreds to thousands of years) [5], making it arduous to identify trends and patterns over a prolonged time frame [6]. Owing to the intricate geological interplay, the variability of seismic activity, the inadequacy of comprehensive data, and the current technological limitations, the prediction of earthquakes remains a complex and formidable field.

Moreover, conventional prediction methods based on empirical (physical or statistical) models are often oversimplified and fallacious when applied to real-life scenarios [7]. With the rapid development of artificial intelligence (AI) in recent years, many research fields have been benefited, including earthquake prediction. At the core of AI lies machine learning, which plays an essential role in driving this transformation. Machine learning's ability to identify the corresponding functional relationships between vast amounts of

data and corresponding labels is its primary advantage. These relationships can be high-dimensional and nonlinear, making it challenging for humans to comprehend, as is the case with earthquake prediction [8,9]. Traditionally, experts' experiences formed the basis of earthquake prediction, which led to random and uncertain outcomes. However, the application of machine learning to earthquake prediction provides a promising approach to achieving more accurate and reliable results.

As early as the 1990s, some scholars proposed the use of machine learning in the area of research in seismology [10]. Currently, various machine learning methods have been applied to seismic classification and location [11–14], seismic event prediction [15–17], seismic early warning [18], seismic exploration [19], slow slip event detection [20], and tomography [21], achieving promising preliminary results in these research fields. Meanwhile, the rapid deployment of long-term seismic monitoring, providing a huge amount of seismic dataset information, accelerates the application of machine learning methods in seismology [8,9,22].

In the research of time series and magnitude prediction, neural networks, including LSTM and convolution neural networks, have been widely used in recent years [23,24]. In [25], the authors proposed a novel approach to earthquake prediction using LSTM networks to capture spatiotemporal correlations among earthquakes in different locations. Their simulation results demonstrate that their method outperforms traditional approaches.

The use of seismicity indicators as inputs in machine learning classifiers has been shown to improve accuracy in earthquake prediction [26]. Results from applying this methodology to four cities in Chile demonstrate that robust predictions can be made by exhaustively exploring how certain parameters should be set up. In [27], a proposed methodology based on the computation of seismic indicators and GP-AdaBoost classification has been trained and tested for three regions: the Hindu Kush, Chile, and Southern California. The obtained prediction results for these regions exhibit improvement when compared with already available studies.

The application of artificial neural networks (ANNs) to earthquake prediction is explored in [28]. The results from the application of ANNs to Chile and the Iberian peninsula are presented, along with a comparative analysis with other well-known classifiers. The conclusion is that the use of a new set of inputs improved all classifiers, but the ANN obtained better results than any other classifier.

A methodology for discovering earthquake precursors by using clustering, grouping, constructing a precursor tree, pattern extraction, and pattern selection has been applied to seven different datasets from three different regions [29]. Results show a remarkable improvement in terms of all evaluated quality measures compared to the former version. The authors suggest that this approach could be further developed and applied to other regions with different geophysical properties to improve earthquake prediction.

In [30], the authors use machine learning techniques to detect signals in a correlation time series corresponding to future large earthquakes. The overall quality is measured by decision thresholds and receiver operating characteristic (ROC) methods together with Shannon information entropy. They hope that the deep learning approach will be more general than previous methods and not require prior guesswork as to what patterns are important.

The seismic activity parameters constructed based on seismic catalogs are typically used as the input data set for earthquake prediction. However, it is still debatable whether the information about small earthquakes (typically with a magnitude smaller than 4.0), such as foreshocks and aftershocks, contained in these seismic features can predict large earthquakes (typically with a magnitude larger than 6.0). In fact, the ability to successfully predict earthquakes has been the subject of controversy [1,3] among researchers. Some studies have shown that information about small earthquakes can indicate the occurrence of large earthquakes, while others have drawn opposite conclusions [31,32].

To clarify this debate, a study was conducted by using seismic features based on seismic catalogs containing small earthquakes to test whether this information can help

predict large earthquakes. Additionally, the possibility of using machine learning in earthquake prediction was explored. The study focused on two important questions:

1. Will there be a strong earthquake ($M \geq 6.0, 7.0$, or 8.0) in the next year?
2. What will be the maximum magnitude of the earthquake in the next year?

The Sichuan–Yunnan region, as shown in Figure 1, was chosen as the research area and seismic features were extracted from the seismic catalog to predict earthquakes.

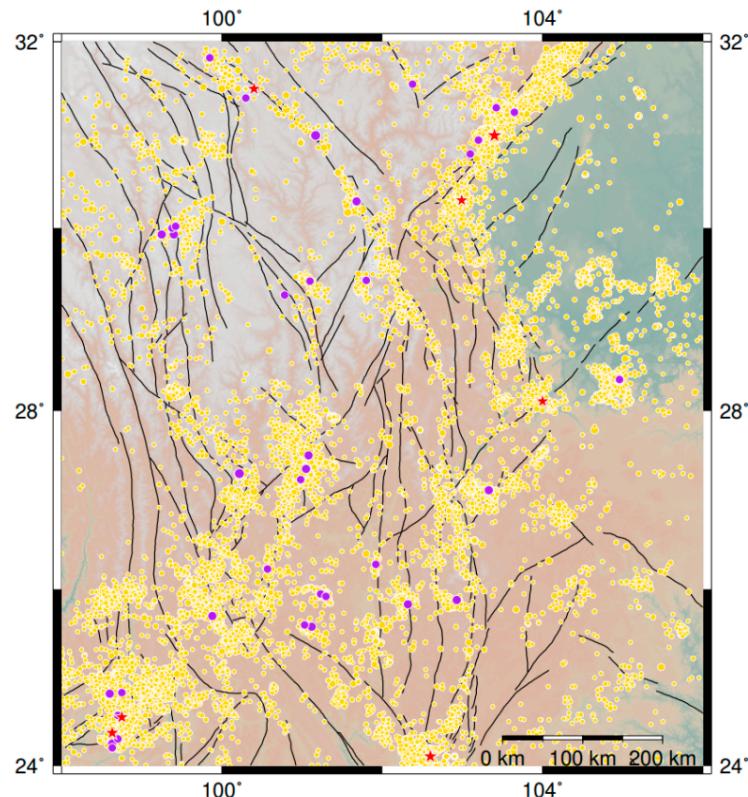


Figure 1. Spatial distribution of seismicity in Chuandian region from 1970 to 2021. The black lines represent active faults [33]. The yellow circles represent earthquakes larger than 3.0, the purple circles show earthquakes larger than 6.0, and the red stars are earthquakes larger than 7.0.

The traditional machine learning methods were used to classify whether there would be strong earthquakes in the next year, and the LSTM network was used to estimate the maximum magnitude of the earthquake in the next year. In this way, we explored the potential applications of machine learning methods in earthquake prediction and provided a possible solution for seismic hazard evaluation.

2. Dataset and Feature Engineering

The seismic catalog used in this study was obtained from the China Earthquake Data Center (CEDC, <http://data.earthquake.cn/index.html>, last accessed on 23 May 2022) and includes earthquake events with a magnitude greater than 3.0 in the Sichuan–Yunnan region from 1970 to 2021. By means of feature engineering, seismic activity parameters were generated based on several statistical laws. These parameters were derived from the seismic catalog and were used as the input features for earthquake prediction, rather than the original catalog itself.

To test these methods, the Chuandian region of Southwestern China (98.0° E– 106.0° E, 24.0° N– 32.0° N) was selected due to its abundant earthquakes. The time range for analysis was from 1 January 1970 to 23 May 2021. While there were several sudden large increases in earthquake activity due to large earthquakes during this time period, the complete-

ness magnitude was estimated to be approximately 3.0 using the maximum curvature technique [34,35] (Figure 2). Therefore, the cutoff magnitude was set at 3.0 for this study.

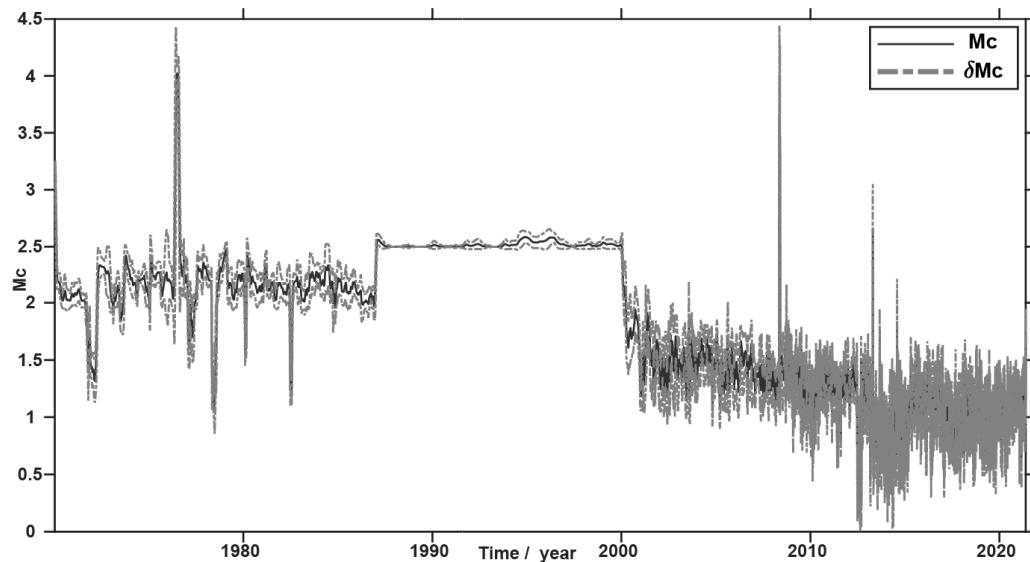


Figure 2. The temporal completeness magnitude from 1970 to 2021 in the Chuandian region based on the maximum curvature technique [34,35].

Prior research has demonstrated that many seismic features derived from earthquake catalogs can be used to predict earthquakes [36–38]. These features include an ‘a’ value and a ‘b’ value in the Gutenberg–Richter law, the number and maximum/mean magnitude of past earthquakes, seismic energy release, magnitude deficit, seismic rate changes, and elapsed time since the last large earthquake. In addition, the standard deviation of the estimated b value, the deviation from the Gutenberg–Richter law, and the probability of earthquake occurrence are also calculated as seismic features. The formulas used to calculate these seismic features are listed in Table 1.

Table 1. Details of seismic features and mathematical expressions.

Seismic Features	Description	Mathematical Expressions
N.O.	Number of earthquakes in observation window	
Mag_max	Maximum magnitude in observation window	$\max\{M_i\}, \text{when } t \in [t_j, t_j + t_{obs}]$
Mag_mean	Mean magnitude in observation window	$\frac{\sum_i M_i}{n}$
b_lsq	b value using least square regression analysis	$\frac{n \sum (M_i \log N_i) - \sum M_i \sum \log N_i}{(\sum M_i)^2 - n \sum M_i^2}$
a_lsq	a value using least square regression analysis	$\frac{\sum (\log N_i + b_{lsq} \cdot M_i)}{n}$
b_std_lsq	Standard deviation of b_lsq	$2.3(b_{lsq})^2 \sqrt{\frac{\sum_{i=1}^n (M_i - Mag_mean)^2}{n(n-1)}}$
std_gr_lsq	Deviation from Gutenberg–Richter law (b_{lsq} , a_{lsq})	$\frac{\sum (\log N_i - a_{lsq} - b_{lsq} \cdot M_i)^2}{n-1}$
b_mlk	b value using maximum likelihood method	$\frac{\log e}{Mag_mean - Mc}$
a_mlk	a value using maximum likelihood method	$\log N + b_{mlk} \cdot M_c$
b_std_mlk	Standard deviation of b_mlk	$2.3(b_{mlk})^2 \sqrt{\frac{\sum_{i=1}^n (M_i - Mag_mean)^2}{n(n-1)}}$

Table 1. Cont.

Seismic Features	Description	Mathematical Expressions
std_gr_mlk	Deviation from Gutenberg–Richter law (b_{mlk} , a_{mlk})	$\frac{\sum(\log N_i - a_{\text{mlk}} - b_{\text{mlk}} \cdot M_i)^2}{n-1}$
dM_lsq	Magnitude deficit (b_{lsq} , a_{lsq})	$Mag_{\text{max}} - a_{\text{lsq}} / b_{\text{lsq}}$
dM_mlk	Magnitude deficit (b_{mlk} , a_{mlk})	$Mag_{\text{max}} - a_{\text{mlk}} / b_{\text{mlk}}$
Energy	Seismic energy release	$\sqrt{\sum(10^{12+1.8M_i})}$
x7_lsq	Probability of earthquake occurrence using b_{lsq}	$e^{\frac{-3b_{\text{lsq}}}{\log e}}$
x7_mlk	Probability of earthquake occurrence using b_{mlk}	$e^{\frac{-3b_{\text{mlk}}}{\log e}}$ $\frac{R_1 - R_2}{\sqrt{\frac{S_1}{n_1} + \frac{S_2}{n_2}}},$ where R_1 and R_2 are seismic rate for the first and second half interval in the observation window. S_1 and S_2 represent the standard deviation of seismic rate R_1 and R_2 . n_1 and n_2 are the number of earthquakes in those two intervals.
zvalue	Seismic rate change	$\frac{M(t, \delta) - n\delta}{\sqrt{n\delta(1-\delta)}},$ where n is the number of earthquakes of the whole seismic catalog, t is the time duration, and δ is the normalized duration of interest. $M(t, \delta)$ represents the observed number of earthquakes by defining end time t and interval of interest δ .
beta	Seismic rate change	
T_elaps6	Days since the last M6.0 earthquake	
T_elaps65	Days since the last M6.5 earthquake	
T_elaps7	Days since the last M7.0 earthquake	
T_elaps75	Days since the last M7.5 earthquake	

The temporal variations of all 22 seismic features listed in Table 1. were calculated for the Chuandian region from 1970 to 2021 using a sliding window process, similarly to previous studies [39]. To predict earthquakes on a mid-term basis, the observation window, the prediction window, and the sliding window were set to be 2 years, 1 year, and 30 days, respectively (Figure 3). This resulted in 591 time steps with 22 seismic features at each step. For earthquake classification, labels were marked with either 0 or 1, and observed magnitudes were used for earthquake magnitude prediction.

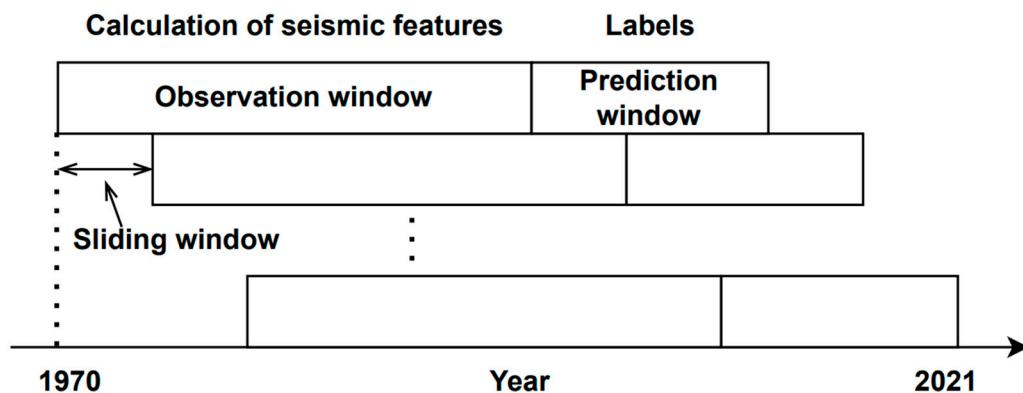


Figure 3. Schematic diagram of seismic feature generation and labels using sliding window approach. The observation window, the prediction window, and the sliding window are set to be 2 years, 1 year, and 30 days, respectively.

3. Methods

In this study, both traditional machine learning algorithms and LSTM neural networks are employed to investigate the occurrence of strong earthquakes and predict their magnitudes, respectively. In the following sections, a brief introduction to these two approaches will be provided. Additionally, the complete earthquake prediction process is illustrated in Figure 4.

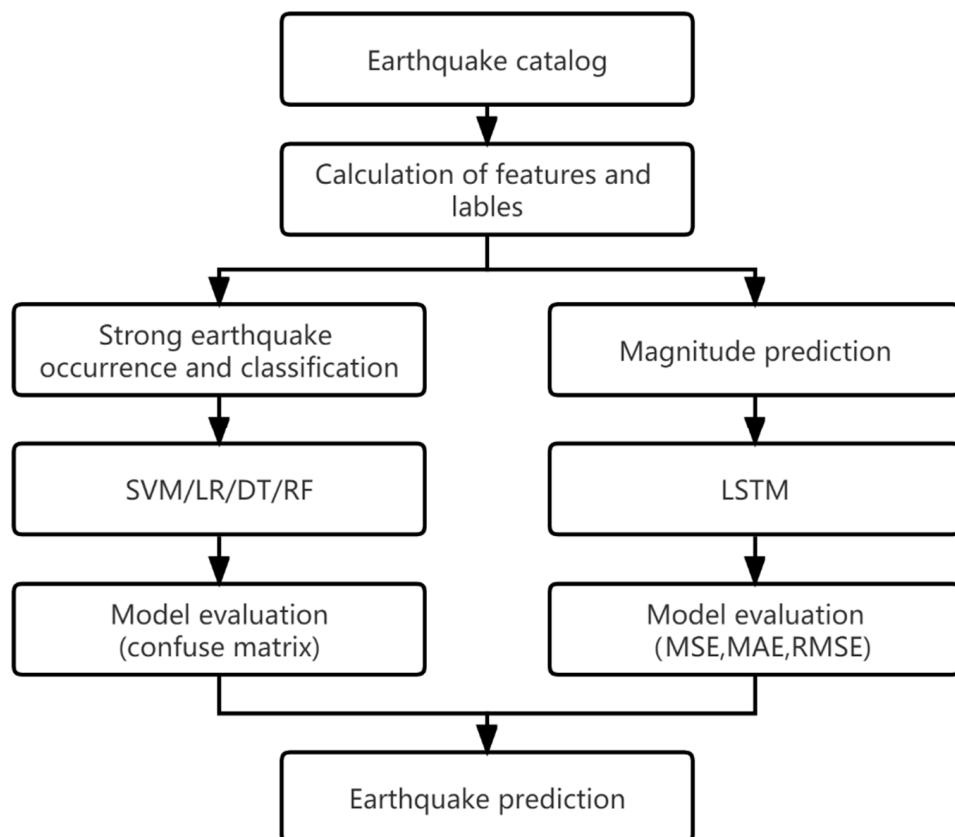


Figure 4. Flow chart of earthquake prediction.

3.1. Traditional Machine Learning Algorithms

Support vector machine (SVM) is a widely used machine learning algorithm for classification and regression problems. It constructs a decision boundary defined by a hyperplane in the feature space, which maximizes the distance between the hyperplane and the nearest training samples, thereby improving the generalization performance of the model. In high-dimensional space, SVM can efficiently handle nonlinear problems, and it performs well for data with small sample size but high dimensionality.

Logistic regression (LR) is a commonly used binary classification algorithm, mainly used to predict the probability of an output variable given an input variable. It calculates the weighted sum of the input variables and passes it through a sigmoid function to map the result to the $[0, 1]$ interval, representing the probability. Logistic regression can use optimization algorithms such as gradient descent for parameter estimation and supports extended forms such as polynomial regression.

Decision tree (DT) is a commonly used machine learning algorithm, which makes decisions by constructing a tree-shaped model. In a decision tree, each node represents a feature, each branch represents a feature value, and each leaf node represents a decision result. By partitioning the data and selecting features, decision trees can effectively perform classification and regression tasks.

Random forest is an ensemble learning algorithm that combines multiple decision trees for classification and regression tasks. In random forests, each decision tree is trained

using randomly selected samples and features, and the final result is obtained by voting or averaging. Due to its ability to reduce overfitting and improve prediction performance, random forest is widely used in various machine learning problems.

3.2. Long Short-Term Memory Neutral Network

LSTM is a special kind of recurrent neural network (RNN) proposed in 1997, mainly to solve the problem of gradient disappearance and gradient explosion in the long time series training process. In comparison to RNNs, LSTM performs better for longer time series data. In recent years, LSTM has been widely used in various fields, such as traffic flow prediction [40], stock yield forecast [41], trajectory prediction [42,43], and earthquake forecasting [37].

The memory unit structure of LSTM, consisting of the forget gate, the input gate, and the output gate, is illustrated in Figure 5. At the current time step, the memory unit takes in the hidden variable h_{t-1} , the memory variable C_{t-1} , and the input x_t . Then, the calculation of the forget gate, the input gate, and the output gate yields the output variables h_t and C_t , which are then fed to the next unit.

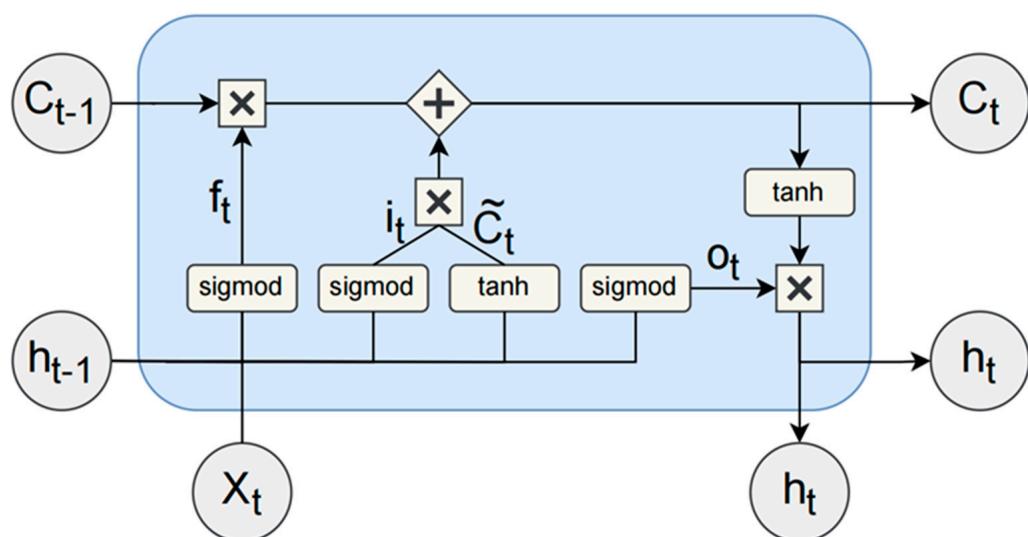


Figure 5. The structure of one memory block in LSTM neural network.

The forget gate first adds h_{t-1} and x_t and passes the result through a sigmoid function to obtain the forget factor, which is then multiplied with C_{t-1} . The forget factor is calculated as follows:

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

Similarly, the input gate firstly adds h_{t-1} and x_t , and passes the result through a sigmoid function and a tanh function to obtain i_t and \tilde{C}_t , respectively. The input gate then multiplies the results of the two functions and adds the results with the output of the forget gate to obtain C_t . The calculation formula of i_t , \tilde{C}_t , and C_t is as follows:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (3)$$

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \quad (4)$$

Finally, the output gate adds h_{t-1} and x_t , and passes the result through a sigmoid function to obtain the forget factor. The cell state is then passed into the tanh function and

multiplied by the forget factor to obtain the new hidden state, which is then passed into the next unit. The calculation formula of o_t and h_t is as follows:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (5)$$

$$h_t = o_t \times \tanh(C_t) \quad (6)$$

Here, W_f , W_i , W_C , and W_o are weight parameters and b_f , b_i , b_C , and b_o are bias parameters.

4. Results

4.1. Evaluation Metrics

The prediction of the occurrence of strong earthquakes involves a two-class classification problem, and the evaluation of its prediction results typically employs a confusion matrix (Table 2).

Table 2. Confusion matrix of binary classification problem.

	Predicted Condition Is Positive	Predicted Condition Is Negative
Actual condition is positive	True Positive (TP)	False Negative (FN)
Actual condition is negative	False Positive (FP)	True Negative (TN)

Specific judgments regarding the performance of earthquake occurrence prediction models are typically made by calculating four evaluation indicators based on the confusion matrix:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (7)$$

$$Precision = \frac{TP}{TP + FP} \quad (8)$$

$$Recall = \frac{TP}{TP + FN} \quad (9)$$

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \quad (10)$$

Additionally, ROC curves and area under curve (AUC) are used to illustrate the relationship between the aforementioned indicators and present the results.

For magnitude prediction, mean square error (MSE), mean absolute error (MAE), and root mean square error ($RMSE$) are calculated to evaluate the prediction accuracy of the model. MSE represents the prediction error, MAE represents the average absolute error between the predicted value and the observed value, and $RMSE$ reflects the degree of deviation between the predicted value and the true value. The formula to calculate each index is as follows:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (11)$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (12)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (13)$$

where n is the number of predicted values, y_i is the true value, and \hat{y}_i is the predicted value.

4.2. Forecast Results

4.2.1. Strong Earthquake Occurrence and Classification

Figure 6 illustrates the evaluation results of four traditional machine learning models (random forest (RF), decision tree (DT), support vector machine (SVM), and logistic regression (LR)) for the classification of large earthquake occurrence, namely, accuracy, precision, recall, and F1 score. The four models were implemented by calling the respective functions of SVM in the sklearn toolkit, LogisticRegression, tree in the ensemble, and RandomForestClassifier in the ensemble. For SVM, the kernel function was set to rbf, and the penalty relaxation variable was set to 1.0, with other parameters adopting default values. Pandas and NumPy were called here to read the dataset file and generate the corresponding array, respectively. Finally, matplotlib.pyplot was used to visualize the predicted results. The four evaluation indicators were calculated using Formulas (7)–(10). All tasks were completed using Python 3.9.6.

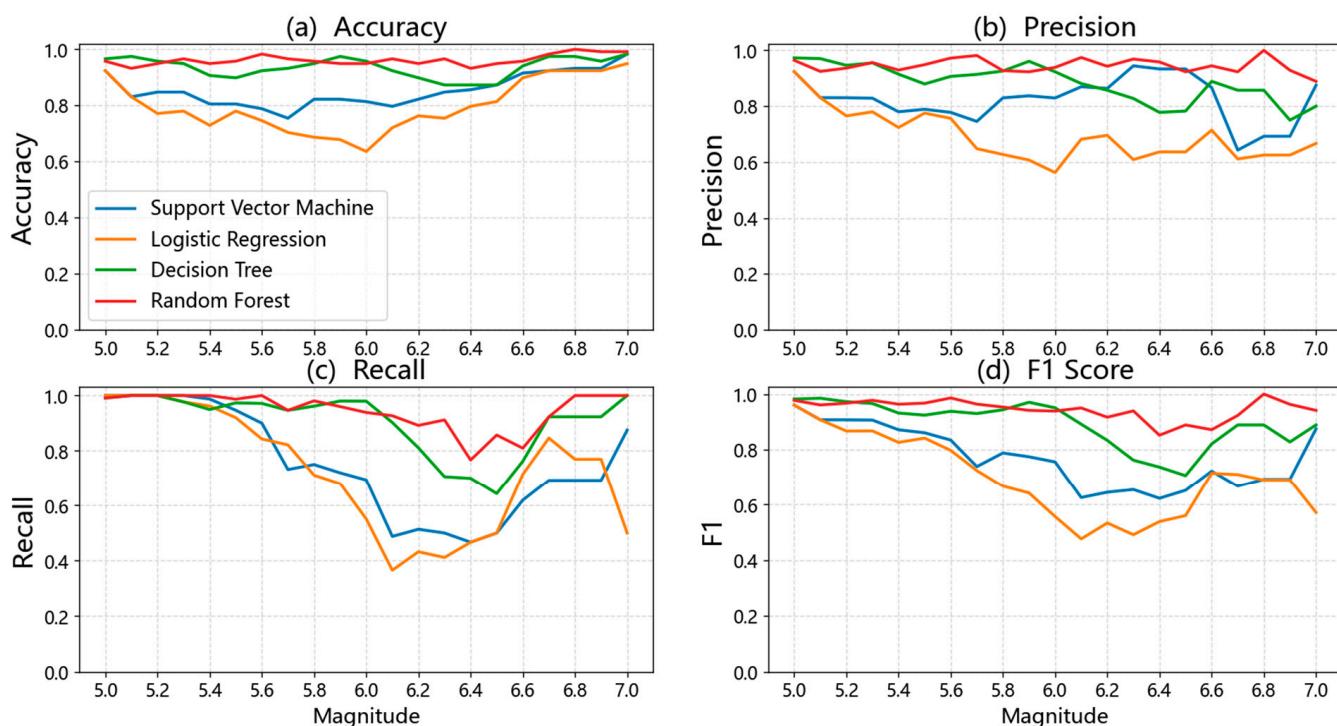


Figure 6. Results of large earthquake occurrence classification: accuracy, precision, recall, and F1 score. A magnitude range of 5.0 to 7.0 is used to represent magnitude threshold of prediction.

The vertical axis of each subgraph in Figure 6 represents the respective evaluation metric value, while the horizontal axis represents the magnitude threshold based on the magnitude range of the catalog. For the experiment's training and test sets, the entire dataset was divided using a 7:3 ratio by calling the function `train_test_split` in the `model_selection` of the `sklearn` package.

During the experiment, it was observed that using the dataset directly as input data for support vector machine and logistic regression resulted in poor classification indicators at certain magnitude thresholds. This was mainly due to the fact that the test data points, which were not of the same class at these thresholds, were only divided into one class. To address this issue, the dataset was standardized and normalized before being used as input. After comparing the experiment's results before and after normalization, it was found that the classification results had been improved to some extent.

Figure 7 displays the ROC curves for the classification of large earthquake occurrences. Unlike the four evaluation indicators, the ROC curve can evaluate the model without requiring a threshold to be set, providing results that better reflect the true performance of

the model. Moreover, the ROC curve remains unaffected even when the distribution proportion of positive and negative samples in the test set changes. This feature is particularly important when dealing with category imbalances in actual datasets, as the ROC curve is able to effectively eliminate the impact of such imbalances on the evaluation results.

An important measure derived from the ROC curve is the area under the curve (AUC). The closer the AUC value is to 1, the better the classification performance of the model. When the AUC is 0.5, the classification result is no better than random guessing. As depicted in Figure 7, the RF classifier achieved the highest AUC value of 0.98, indicating its superior performance in earthquake prediction. In contrast, the LR classifier had the lowest AUC value of 0.72, indicating the weakest performance among the four classifiers.

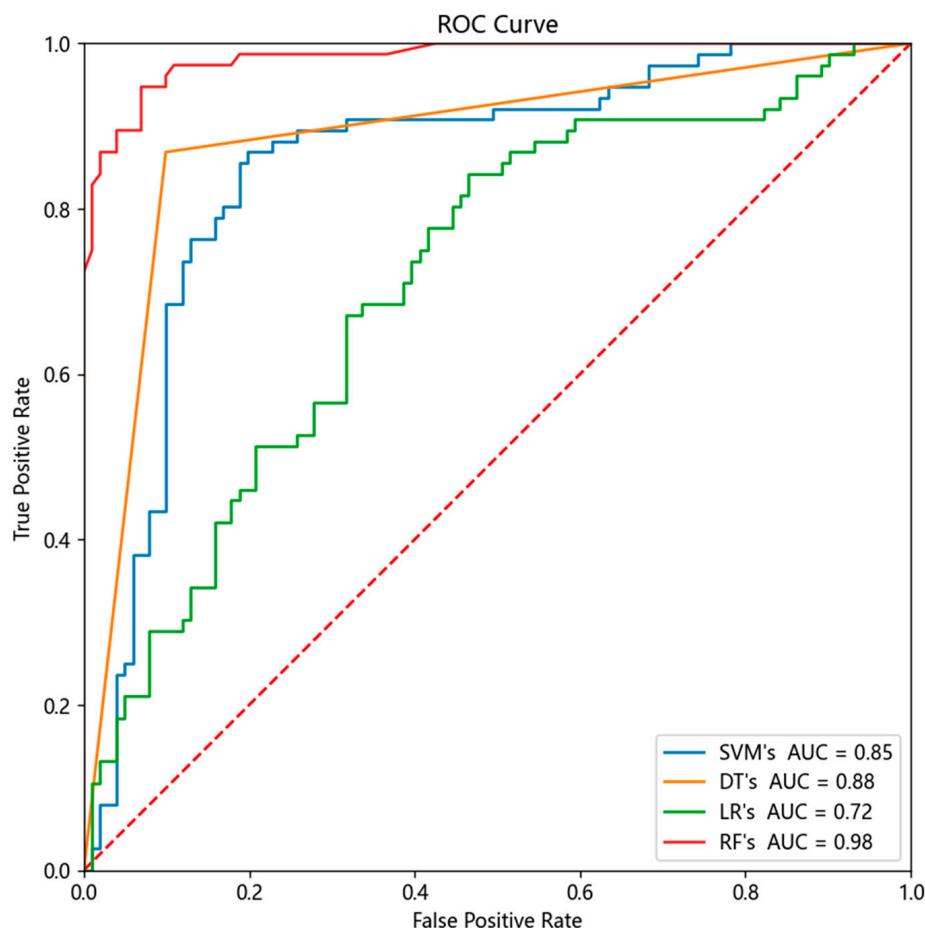


Figure 7. ROC curves of the large earthquake occurrence classification using SVM, DT, LR, RF.

4.2.2. Magnitude Prediction

In the magnitude prediction process, the same 22 feature parameters were utilized as input to the LSTM model, which then outputted the maximum magnitude of the next forecast window. The training set, validation set, and test set were divided in an 8:1:1 ratio. Given the small size of the data set, we needed to be cautious of overfitting. Therefore, we performed multiple optimizations using the validation set and determined the optimal configuration of the LSTM model. Specifically, the number of hidden layers was set as 1, the number of neurons as 16, the initial learning rate as 0.01, and the number of epochs as 200, to prevent overfitting. Pandas and NumPy were also called here to read the dataset file and generate the corresponding array, respectively. Torch was imported to build the LSTM model. All figures were obtained using matplotlib.

In the data preprocessing stage, the MinMaxScaler function was utilized to normalize the data, which was then fed into the model for training. The denormalized prediction results were obtained after the model had made its predictions. The MinMaxScaler function

of sklearn was called in this section. The magnitude prediction results are presented in Figures 8 and 9.

Figures 8 and 9 present the outcomes of the maximum magnitude prediction. The LSTM model can effectively capture the temporal variations of the maximum magnitudes in the training and validation sets, with most of the predicted magnitudes oscillating within a range of ± 0.5 of the observed magnitude. However, on the test set, the model can only grasp the trend of the magnitude and tends to overestimate the maximum magnitude for actual events with magnitude $<= 5.0$ and underestimate the maximum magnitude for actual events with magnitude $>= 6.0$. This suggests that although the LSTM model can detect the general pattern, it tends to produce oversimplified predictions for the maximum magnitude.

Figures 10 and 11 illustrate the dispersion of errors using boxplot and histogram. Notably, the error distribution of the training set is centered around 0, whereas the error distribution of the test set is much wider, centered around 0.5. In the test set, the model produces a higher quantity of positive errors than negative ones, as is evident from the histograms in Figure 11.

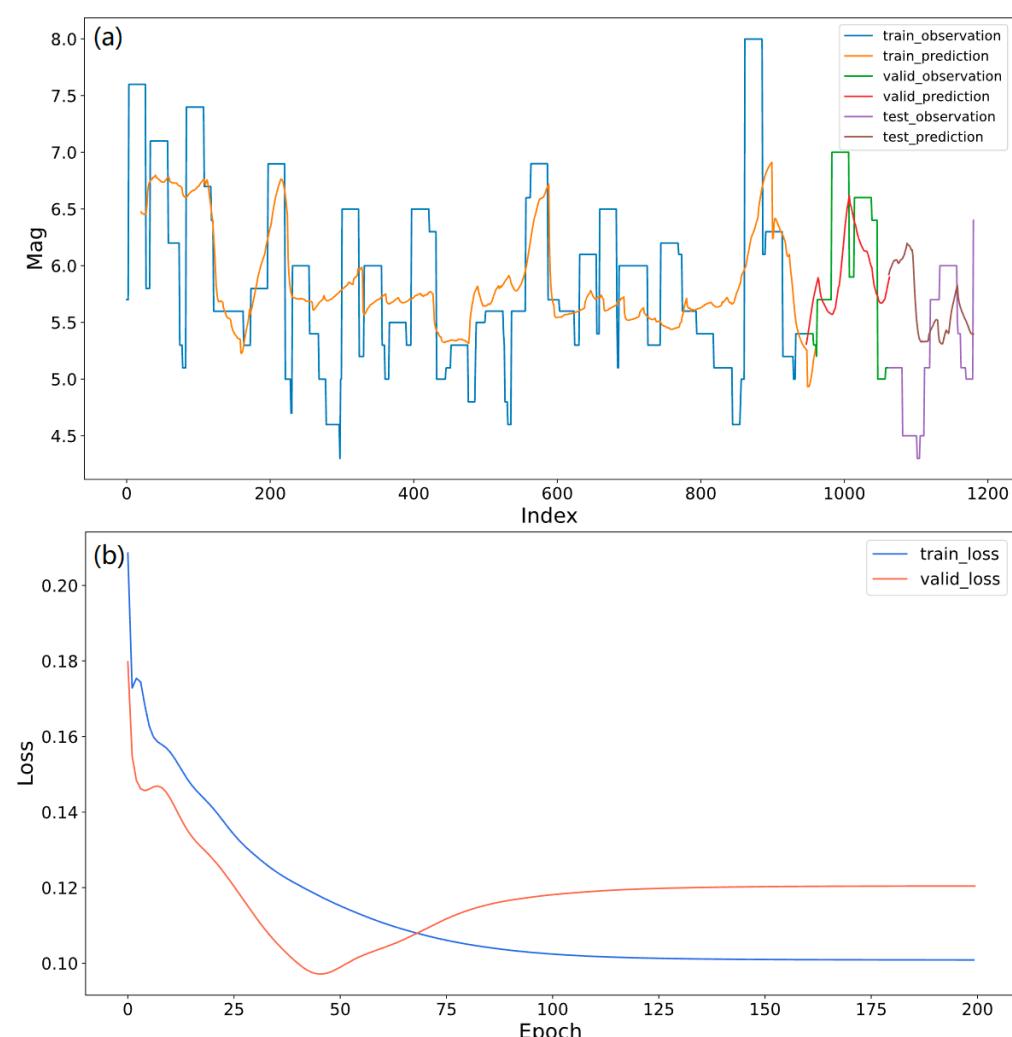


Figure 8. (a) Retrospective test of prediction of the maximum magnitude earthquake. The blue, green, and purple curves represent training, validation, and test period of the observations, respectively. The yellow, red, and brown curves represent training, validation, and test period of the predictions, respectively. (b) Loss curves of prediction of the maximum magnitude earthquake. The blue and red curves represent the loss of training set and test set, respectively.

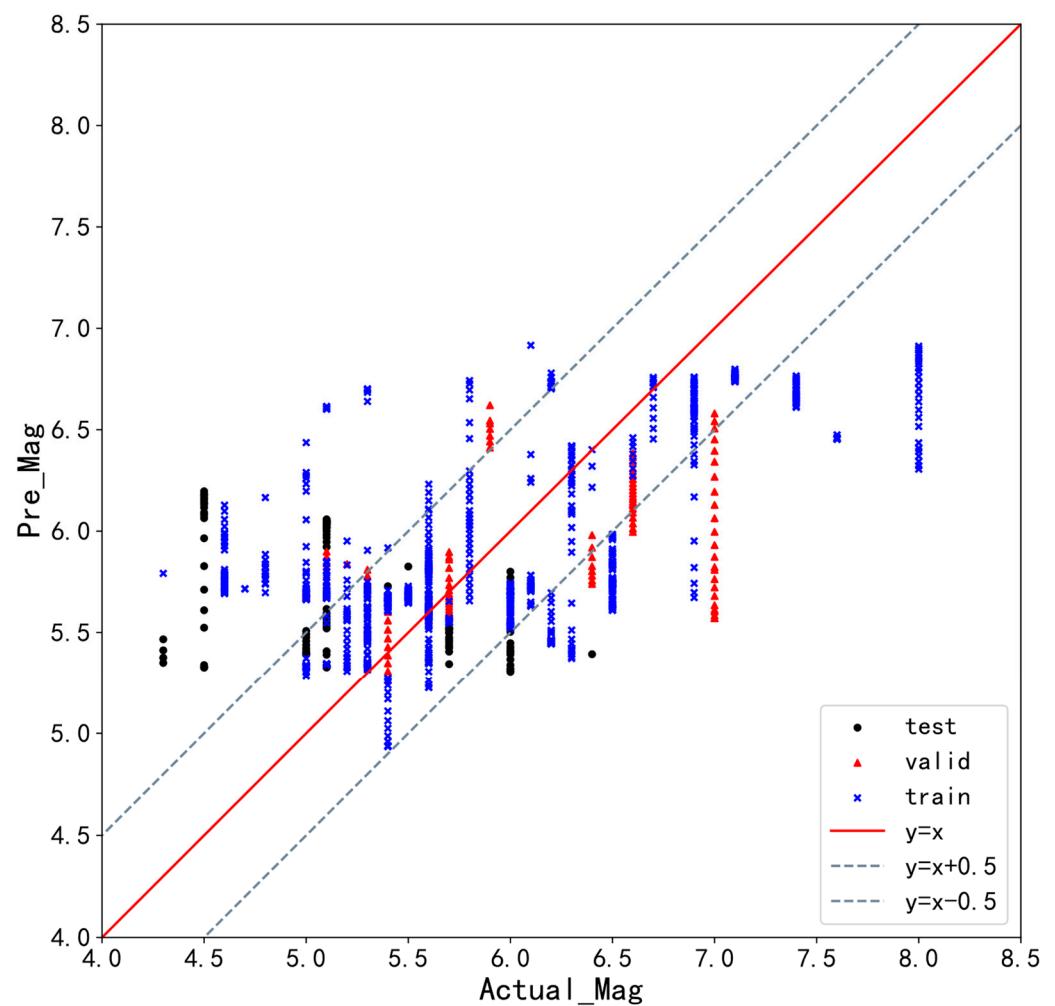


Figure 9. Comparison of the predicted and observed maximum magnitudes. The black dots, red triangles, and blue crosses represent test, validation, and training dataset, respectively.

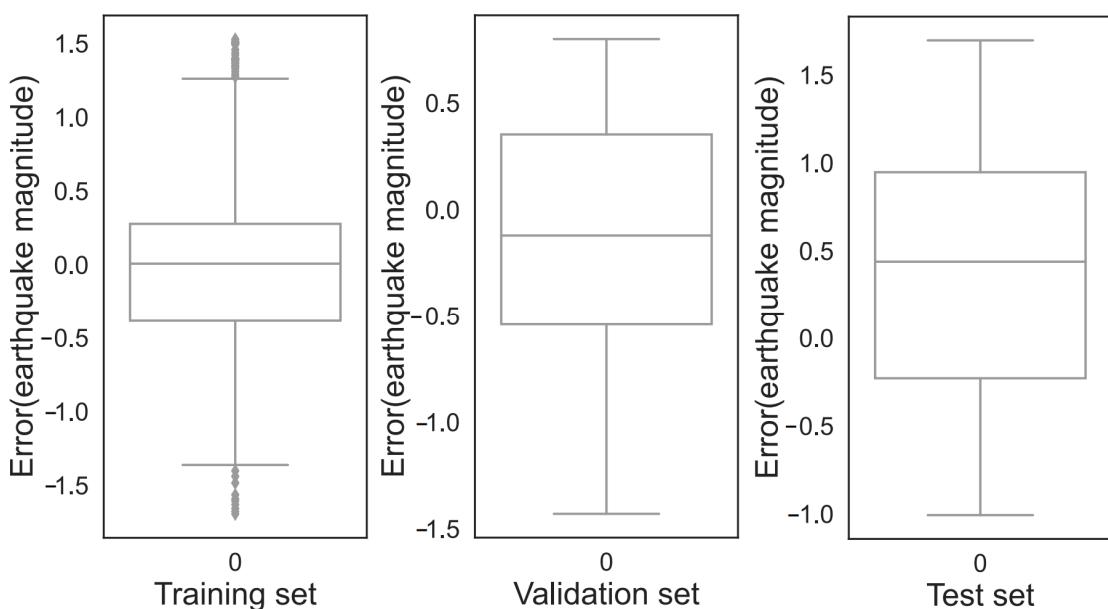


Figure 10. Boxplot of the errors of training set, validation set, and test set.

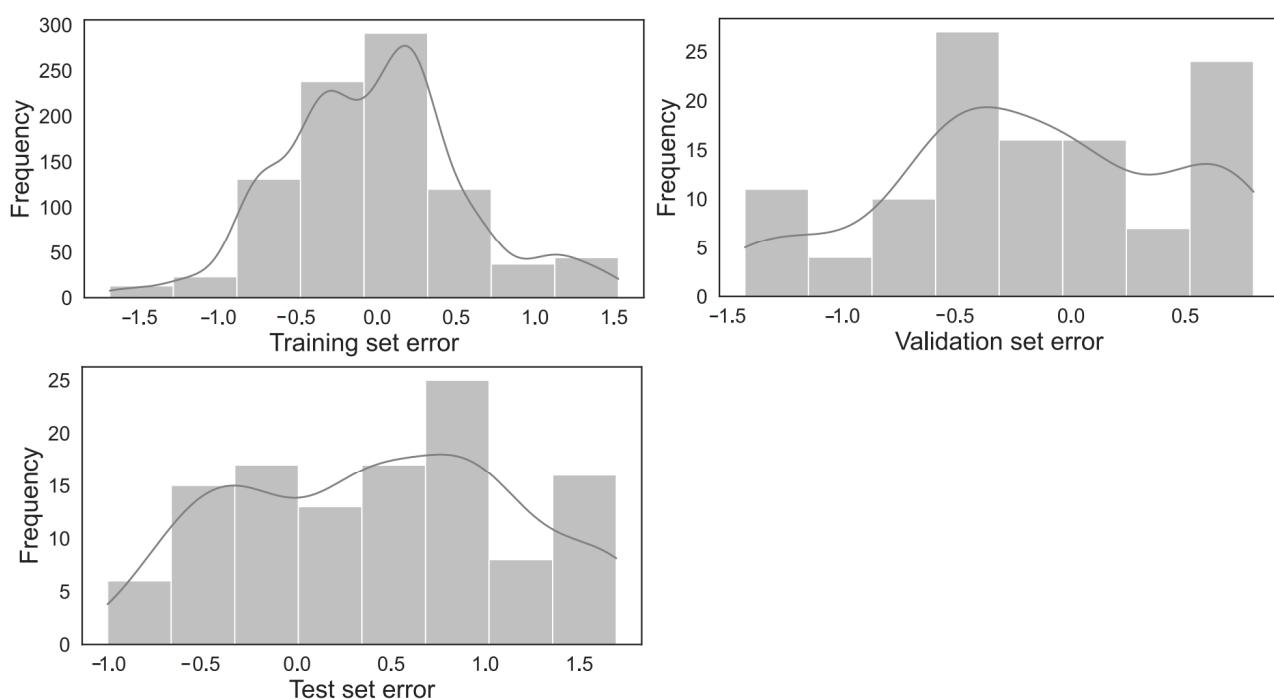


Figure 11. Histograms of the errors of training set, validation set, and test set in earthquake magnitude prediction using LSTM.

Table 3 presents the evaluation indicators of our model and compares them with those of other studies. The results are better than that of [36], but worse than [37]. The reasons for these differences will be discussed in the Section 5.

Table 3. Comparison of prediction results by different researchers.

Evaluation Index	This Study	[37]	[36]		
			DL	RF	GLM
MSE	0.7347	0.015192			
MAE	0.7252	0.097173	1.15	0.74	1.03
RMSE	0.8571	0.123256			

To evaluate the importance of different features, the permutation importance method was used. This method measures the importance of features by calculating the increase of model prediction error after shuffling the time series of each feature. The advantage of this method is that it can compare the importance of different features and save more time compared to other methods.

Figure 12 shows the feature importance of our model, where the length of the bar chart represents the error of the model after shuffling the order. Longer bars indicate more important features, while the orange line represents the MAE of the model as the reference line.

Figure 12 reveals that nearly all the features used have a positive effect on the model's performance. Among them, b_std_mlk, x7_mlk, and T_elaps7 are less important, while dM_mlk, x7_lsq, N.O., and T_elaps6 are the top four most important features. These results demonstrate that magnitude deficit, probability of earthquake occurrence, number of earthquakes, and the days since the last large earthquake are crucial factors for earthquake magnitude prediction, which is consistent with physical understanding.

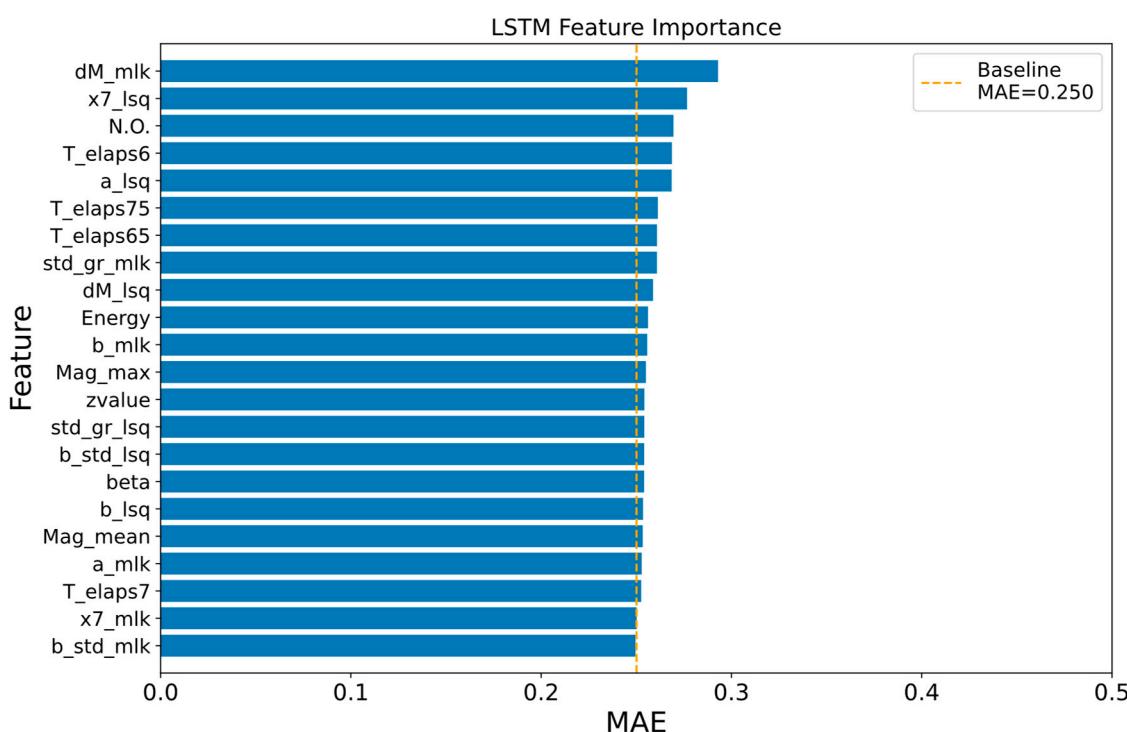


Figure 12. Feature importance obtained by the permutation importance method.

5. Discussion

5.1. Sample Number Issue and Feature Importance

Based on the classification results presented above, it is evident that RF outperforms LR and SVM in terms of several evaluation indicators and is also marginally better than DT. However, it should be noted that the classification results may be affected by the different proportions of positive and negative samples in the dataset, which is determined by the setting of the strong earthquake threshold and can impact the prediction outcomes. To investigate this further, we experimented with several magnitudes which were shown in Table 4 as the classification threshold and analyzed the number of positive and negative samples, as well as the difference in classification metrics. It is important to exercise caution when interpreting the accuracy score in cases where there is an imbalance in the number of positive and negative samples. Newer models that are insensitive to class imbalance [44] may help overcome this problem, but this is out of the scope of this study.

Table 4. Evaluation results and sample numbers of the random forest method.

Magnitude	Accuracy	Precision	Recall	F1 Score	Positive Samples	Negative Samples
5.0	0.975	0.982	0.991	0.986	549	41
5.5	0.958	0.949	0.987	0.967	386	204
6.0	0.949	0.922	0.959	0.940	258	332
6.5	0.958	0.96	0.857	0.906	141	449
7.0	0.992	0.889	1.0	0.941	60	530

At the same time, the impact of the small number of overall samples on the classification results should also be taken into consideration. Although the evaluation metric of RF is relatively the highest among the four classification methods, indicating that it is the best classifier in the prediction of strong earthquakes, further verification is necessary to determine its reliability on different datasets due to the small number of samples in the test set and the uneven distribution of positive and negative samples. To better interpret the classification results of RF, the feature quantity was ranked from high to low based on

their weights in the classification, as illustrated in Figure 13. The top three most important features are T_elaps75, T_elaps7, and dM_lsq, which is consistent with the feature importance of earthquake magnitude prediction and confirms the important role of the number of days since the last large earthquake and the magnitude deficit.

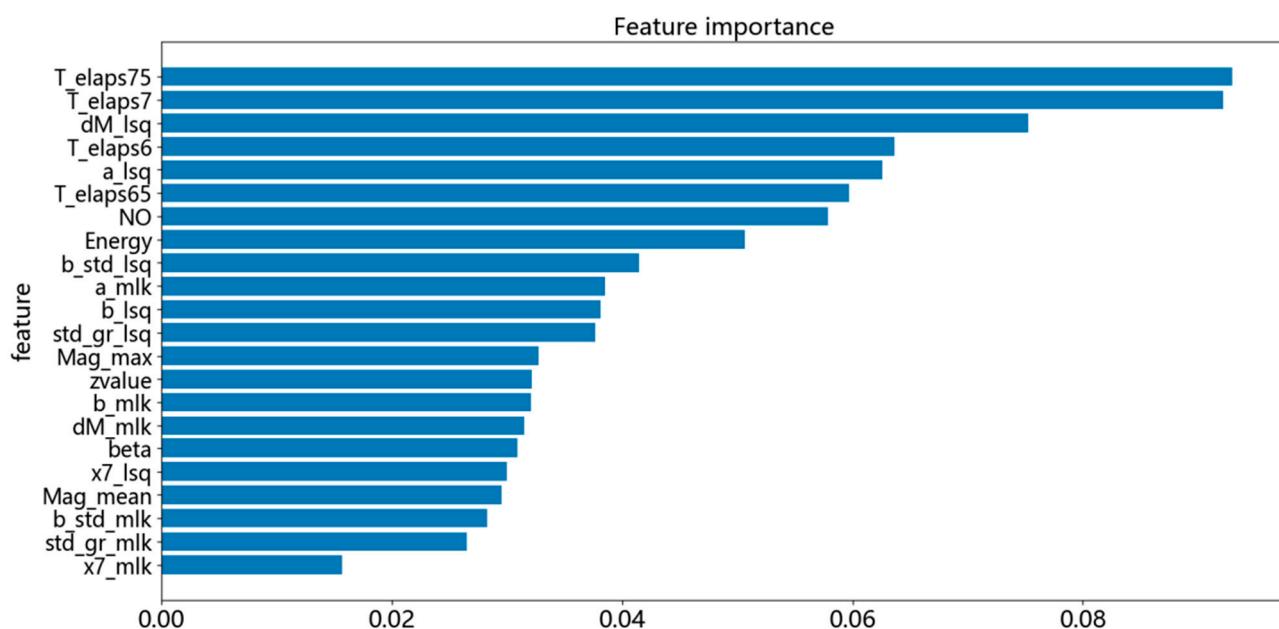


Figure 13. Feature importance ranking by the random forest method.

The poor classification results of LR and SVM in this experiment also suggest the need for further investigation into the potential impact of dataset size and feature distribution on the classification performance. To address this issue, the feature values were standardized and normalized in the dataset, and the classifier parameters were adjusted accordingly. This approach led to an improvement in the overall classification performance.

5.2. Comparison with Previous Studies

From the prediction results, it can be seen that the LSTM model only trended in the prediction results, but the predicted values were significantly larger than the observed values. Upon comparison with two other articles in Table 3, the results were inferior to those reported in [37]. However, after conducting replication experiments, we discovered that this disparity could be attributed to differences in the input features used in those two studies. Using the same input approach as in [37], a similar performance was obtained, as shown in Figures 14 and 15.

In the replication experiment, the MSE was found to be 0.2289, while MAE was 0.4193, and RMSE was 0.4784. The prediction results obtained using the approach of [29] are shown in Figures 14 and 15 and appear to be better than the results obtained previously. However, we identified a potential data leakage issue in their approach due to the overlap between input features and output labels. Specifically, the feature “Mag_max_obs” dominated other features, as confirmed by the feature importance shown in Figure 16. Upon careful examination of the approach of [37], we found that they input the maximum magnitude of the first few windows for training and then obtain the maximum magnitude for the several windows that follow. As these windows have overlapping parts, the maximum magnitudes of the first several windows contain some characteristics of the maximum magnitudes of the following several windows. This data leakage problem resulted in the information of the test set being leaked to the training set, leading to a too-low error and a too-high feature importance of Mag_max_obs.

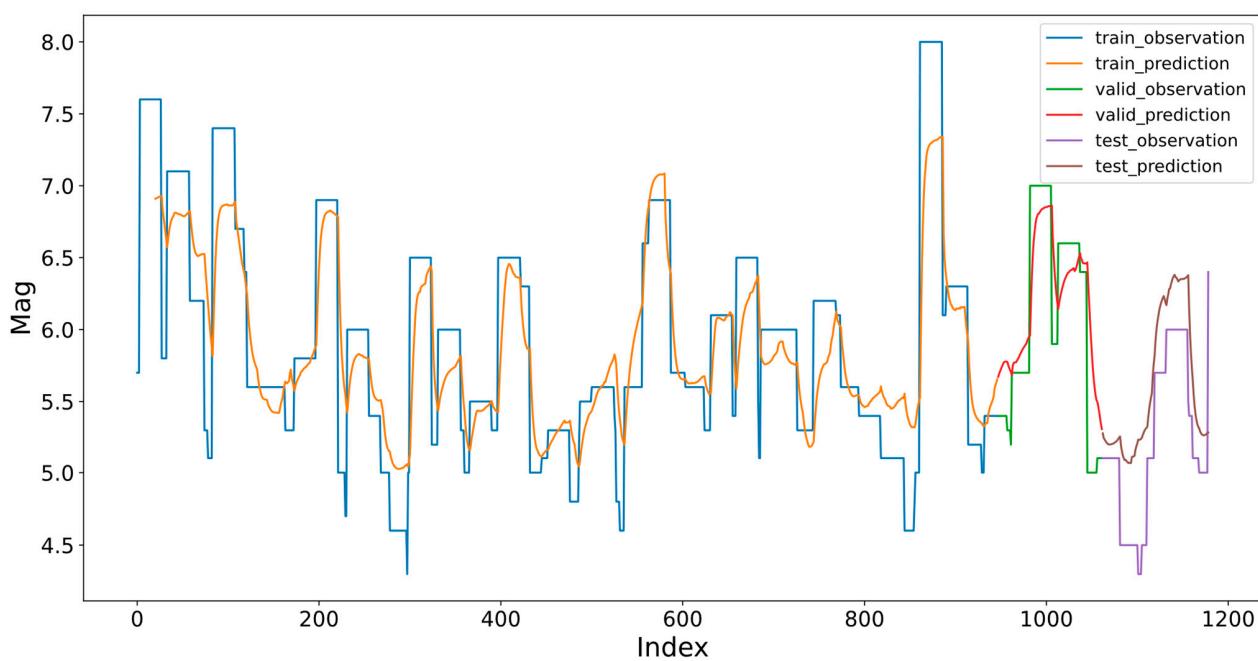


Figure 14. Retrospective test of prediction of the maximum magnitude earthquake using the similar input with [37]. The blue, green, and purple curves represent training, validation, and test period of the observations, respectively. The yellow, red, and brown curves represent training, validation, and test period of the predictions, respectively.

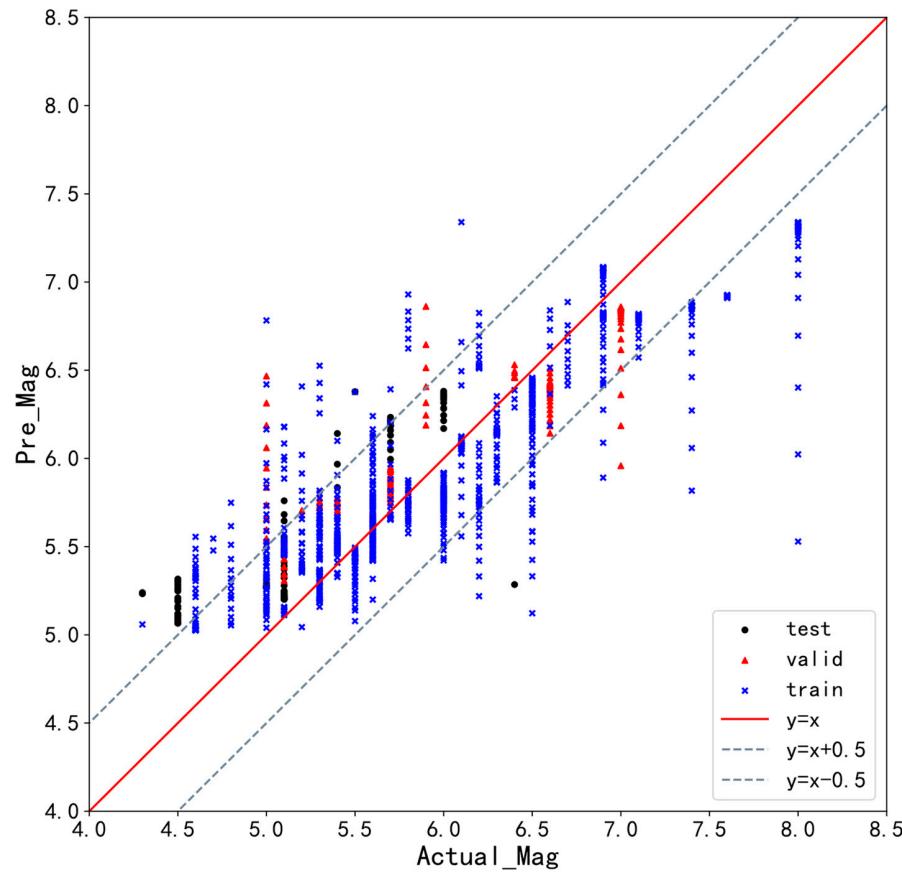


Figure 15. Comparison of the predicted and observed maximum magnitudes using the same input with [37]. The black dots, red triangles, and blue crosses represent test, validation, and training data set, respectively.

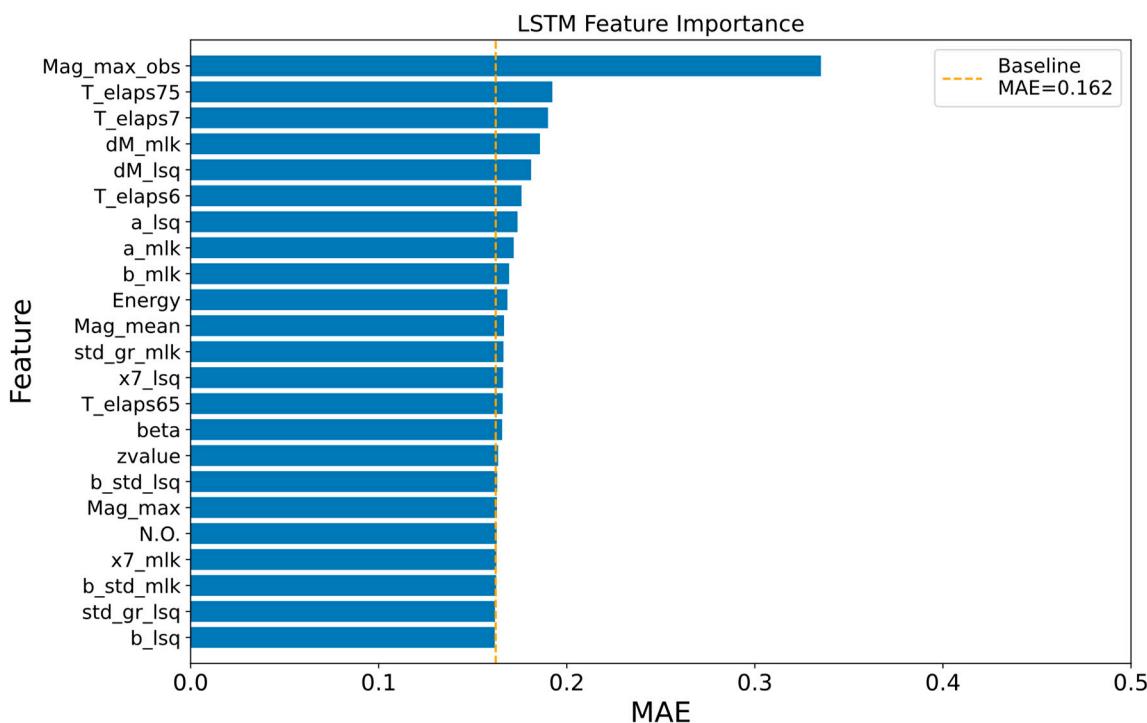


Figure 16. Feature importance for the retrospective experiment with [37].

Meanwhile, in Table 3, a comparative analysis of the results with those presented in [36] is conducted. The authors of this paper employ generalized linear models (GLM), random forest (RF), and deep neural network (DNN) to predict the maximum magnitude of future earthquakes. Unlike this study, their work benefits from sufficient training data. Remarkably, their predicted magnitudes are smaller than the observed values. Notably, the LSTM model outperforms the predictions reported in [36], underscoring the effectiveness of the model construction.

6. Conclusions

In this study, we explored the possibility of using machine learning for earthquake prediction by applying four traditional machine learning methods and the LSTM neural network to predict the occurrences and maximum magnitudes of earthquakes in the Sichuan–Yunnan region. We calculated and extracted seismicity parameters related to earthquake occurrence from the earthquake catalog as input features. The results showed that the random forest method was the most effective at classifying large earthquake occurrences, and the LSTM method provided a reasonable estimation of earthquake magnitude.

The findings support the idea that small earthquakes contain information relevant to predicting future large earthquakes and offer a promising way to predict the occurrence of large earthquakes. Additionally, the findings provide useful information on which features that are consistent with physical interpretation are important for earthquake prediction.

While the limitations of this study should be noted, they also represent the next steps for future work. First, under this framework, earthquake swarms, which are statistically very rare [45], are difficult to predict due to the small differences between their magnitudes. Second, longer-term seismic monitoring is needed for the further application of more complex models (e.g., transformer) to improve the performance of predictions. Third, the spatial locations of earthquakes are not considered in this study but are important for earthquake prediction. New models which can address spatial information (e.g., graph neural networks) may be useful to tackle this problem in the future. Although limitations and difficulties exist, we are trying to explore the nonlinear relations of earthquake prediction, which is one of the most difficult problems in seismology, by applying machine learning

methods. This study presents a potential avenue for improving the accuracy of earthquake prediction in the future.

Author Contributions: Methodology, X.W., Z.Z. and K.J.; Software, X.W., Z.Z., Y.Y. and K.J.; Validation, X.W., Z.Z. and Z.L.; Formal analysis, S.Z., C.J. and K.J.; Resources, K.J.; Data curation, C.J.; Writing—original draft, X.W. and Z.Z.; Writing—review & editing, S.Z. and K.J.; Visualization, X.W. and Z.Z.; Supervision, K.J.; Project administration, K.J.; Funding acquisition, S.Z. and K.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Special Fund of the Institute of Geophysics, China Earthquake Administration (Grant No. DQJB 22Z01-09), the National Natural Science Foundation of China (Grant Nos. 42274068, U2039204), the Key Research and Development Project in Shaanxi Province (Grant No. 2023-YBSF-237), and the Shanghai Sheshan National Geophysical Observatory (Grant No. SSOP202207).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset of this research can be found from the China Earthquake Data Center (CEDC, <http://data.earthquake.cn/index.html>, last accessed on 23 May 2022).

Acknowledgments: We benefit from helpful discussions with Naidan Yun, Han Yue and Xiaoyan Cai. Comments and suggestions from Editor and two anonymous reviewers are appreciated.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Geller, R.J.; Jackson, D.D.; Kagan, Y.Y.; Mulargia, F. Earthquakes cannot be predicted. *Science* **1997**, *275*, 1616. [[CrossRef](#)]
2. Knopoff, L. Earthquake prediction: The scientific challenge. *Proc. Natl. Acad. Sci. USA* **1996**, *93*, 3719–3720. [[CrossRef](#)]
3. Wyss, M. Cannot earthquakes be predicted? *Science* **1997**, *278*, 487–490. [[CrossRef](#)]
4. Sibson, R.H. Crustal stress, faulting and fluid flow. *Geol. Soc. Lond. Spec. Publ.* **1994**, *78*, 69–84. [[CrossRef](#)]
5. Liu, M.; Stein, S. Mid-continental earthquakes: Spatiotemporal occurrences, causes, and hazards. *Earth-Sci. Rev.* **2016**, *162*, 364–386. [[CrossRef](#)]
6. Field, E.H. How Physics-Based Earthquake Simulators Might Help Improve Earthquake Forecasts. *Seismol. Res. Lett.* **2019**, *90*, 467–472. [[CrossRef](#)]
7. Shi, Y.; Sun, Y.; Luo, G.; Dong, P.; Zhang, H. Roadmap for earthquake numerical forecasting in China—Reflection on the tenth anniversary of Wenchuan earthquake. *Chin. Sci. Bull.* **2018**, *63*, 1865–1881. [[CrossRef](#)]
8. Bergen, K.J.; Johnson, P.A.; de Hoop, M.V.; Beroza, G.C. Machine learning for data-driven discovery in solid Earth geoscience. *Science* **2019**, *363*, eaau0323. [[CrossRef](#)]
9. Beroza, G.C.; Segou, M.; Mostafa Mousavi, S. Machine learning and earthquake forecasting—next steps. *Nat. Commun.* **2021**, *12*, 4761. [[CrossRef](#)]
10. Shimshoni, Y.; Intrator, N. Classification of seismic signals by integrating ensembles of neural networks. *IEEE Trans. Signal Process.* **1998**, *46*, 1194–1201. [[CrossRef](#)]
11. Li, Z.; Meier, M.-A.; Hauksson, E.; Zhan, Z.; Andrews, J. Machine Learning Seismic Wave Discrimination: Application to Earthquake Early Warning. *Geophys. Res. Lett.* **2018**, *45*, 4773–4779. [[CrossRef](#)]
12. Malfante, M.; Dalla Mura, M.; Metaxian, J.-P.; Mars, J.I.; Macedo, O.; Inza, A. Machine Learning for Volcano-Seismic Signals Challenges and perspectives. *IEEE Signal Process. Mag.* **2018**, *35*, 20–30. [[CrossRef](#)]
13. Tang, L.; Zhang, M.; Wen, L. Support Vector Machine Classification of Seismic Events in the Tianshan Orogenic Belt. *J. Geophys. Res. Solid Earth* **2020**, *125*, e2019JB018132. [[CrossRef](#)]
14. Titos, M.; Bueno, A.; Garcia, L.; Benitez, C. A Deep Neural Networks Approach to Automatic Recognition Systems for Volcano-Seismic Events. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2018**, *11*, 1533–1544. [[CrossRef](#)]
15. Asim, K.M.; Martinez-Alvarez, F.; Basit, A.; Iqbal, T. Earthquake magnitude prediction in Hindukush region using machine learning techniques. *Nat. Hazards* **2017**, *85*, 471–486. [[CrossRef](#)]
16. Mousavi, S.M.; Beroza, G.C. A Machine-Learning Approach for Earthquake Magnitude Estimation. *Geophys. Res. Lett.* **2020**, *47*, e2019GL085976. [[CrossRef](#)]
17. Rouet-Leduc, B.; Hulbert, C.; Lubbers, N.; Barros, K.; Humphreys, C.J.; Johnson, P.A. Machine Learning Predicts Laboratory Earthquakes. *Geophys. Res. Lett.* **2017**, *44*, 9276–9282. [[CrossRef](#)]
18. Li, Z.; Tian, K.; Wang, F.; Zheng, X.; Wang, F. Home damage estimation after disasters using crowdsourcing ideas and Convolutional Neural Networks. In Proceedings of the 5th International Conference on Measurement, Instrumentation and Automation (ICMIA), Shenzhen, China, 17–18 September 2016; pp. 857–860.

19. Shahnas, M.H.; Yuen, D.A.; Pysklywec, R.N. Inverse Problems in Geodynamics Using Machine Learning Algorithms. *J. Geophys. Res. Solid Earth* **2018**, *123*, 296–310. [[CrossRef](#)]
20. Provost, F.; Hibert, C.; Malet, J.P. Automatic classification of endogenous landslide seismicity using the Random Forest supervised classifier. *Geophys. Res. Lett.* **2017**, *44*, 113–120. [[CrossRef](#)]
21. Araya-Polo, M.; Jennings, J.; Adler, A.; Dahlke, T. Deep-learning tomography. *Lead. Edge* **2018**, *37*, 58–66. [[CrossRef](#)]
22. DeVries, P.M.R.; Viegas, F.; Wattenberg, M.; Meade, B.J. Deep learning of aftershock patterns following large earthquakes. *Nature* **2018**, *560*, 632–634. [[CrossRef](#)] [[PubMed](#)]
23. Moustra, M.; Avraamides, M.; Christodoulou, C. Artificial neural networks for earthquake prediction using time series magnitude data or Seismic Electric Signals. *Expert Syst. Appl.* **2011**, *38*, 15032–15039. [[CrossRef](#)]
24. Panakkat, A.; Adeli, H. Neural network models for earthquake magnitude prediction using multiple seismicity indicators. *Int. J. Neural Syst.* **2007**, *17*, 13–33. [[CrossRef](#)] [[PubMed](#)]
25. Wang, Q.; Guo, Y.; Yu, L.; Li, P. Earthquake Prediction Based on Spatio-Temporal Data Mining: An LSTM Network Approach. *IEEE Trans. Emerg. Top. Comput.* **2020**, *8*, 148–158. [[CrossRef](#)]
26. Asencio-Cortes, G.; Martinez-Alvarez, F.; Morales-Esteban, A.; Reyes, J. A sensitivity study of seismicity indicators in supervised learning to improve earthquake prediction. *Knowl.-Based Syst.* **2016**, *101*, 15–30. [[CrossRef](#)]
27. Asim, K.M.; Idris, A.; Iqbal, T.; Martinez-Alvarez, F. Seismic indicators based earthquake predictor system using Genetic Programming and AdaBoost classification. *Soil Dyn. Earthq. Eng.* **2018**, *111*, 1–7. [[CrossRef](#)]
28. Martinez-Alvarez, F.; Reyes, J.; Morales-Esteban, A.; Rubio-Escudero, C. Determining the best set of seismicity indicators to predict earthquakes. Two case studies: Chile and the Iberian Peninsula. *Knowl.-Based Syst.* **2013**, *50*, 198–210. [[CrossRef](#)]
29. Florido, E.; Asencio Cortes, G.; Aznarte, J.L.; Rubio-Escudero, C.; Martinez-Alvarez, F. A novel tree-based algorithm to discover seismic patterns in earthquake catalogs. *Comput. Geosci.* **2018**, *115*, 96–104. [[CrossRef](#)]
30. Rundle, J.B.; Donnellan, A.; Fox, G.; Crutchfield, J.P. Nowcasting Earthquakes by Visualizing the Earthquake Cycle with Machine Learning: A Comparison of Two Methods. *Surv. Geophys.* **2022**, *43*, 483–501. [[CrossRef](#)]
31. Rundle, J.B.; Donnellan, A.; Fox, G.; Ludwig, L.G.; Crutchfield, J. Does the Catalog of California Earthquakes, With Aftershocks Included, Contain Information About Future Large Earthquakes? *Earth Space Sci.* **2023**, *10*, e2022EA002521. [[CrossRef](#)]
32. Alexandridis, A.; Chondrodima, E.; Efthimiou, E.; Papadakis, G.; Vallianatos, F.; Triantis, D. Large Earthquake Occurrence Estimation Based on Radial Basis Function Neural Networks. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 5443–5453. [[CrossRef](#)]
33. Deng, Q.D.; Zhang, P.Z.; Ran, Y.K.; Yang, X.P.; Min, W.; Chu, Q.Z. Basic characteristics of active tectonics of China. *Sci. China Ser. D-Earth Sci.* **2003**, *46*, 356–372.
34. Wiemer, S. A Software Package to Analyze Seismicity: ZMAP. *Seismol. Res. Lett.* **2001**, *72*, 373–382. [[CrossRef](#)]
35. Wiemer, S.; Wyss, M. Minimum magnitude of completeness in earthquake catalogs: Examples from Alaska, the western United States, and Japan. *Bull. Seismol. Soc. Am.* **2000**, *90*, 859–869. [[CrossRef](#)]
36. Asencio-Cortes, G.; Morales-Esteban, A.; Shang, X.; Martinez-Alvarez, F. Earthquake prediction in California using regression algorithms and cloud-based big data infrastructure. *Comput. Geosci.* **2018**, *115*, 198–210. [[CrossRef](#)]
37. Li, L.; Shi, Y.; Cheng, S. Exploration of long short-term memory neural network in intermediate earthquake forecast: A case study in Sichuan-Yunnan region. *Chin. J. Geophys. Chin. Ed.* **2022**, *65*, 12–25. [[CrossRef](#)]
38. Reyes, J.; Morales-Esteban, A.; Martinez-Alvarez, F. Neural networks to predict earthquakes in Chile. *Appl. Soft Comput.* **2013**, *13*, 1314–1328. [[CrossRef](#)]
39. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)]
40. Fu, R.; Zhang, Z.; Li, L. Using LSTM and GRU Neural Network Methods for Traffic Flow Prediction. In Proceedings of the 31st Youth Academic Annual Conference of Chinese-Association-of-Automation (YAC), Wuhan, China, 11–13 November 2016; pp. 324–328.
41. Chen, K.; Zhou, Y.; Dai, F. A LSTM-based method for stock returns prediction: A case study of China stock market. In Proceedings of the 2015 IEEE international conference on big data (big data), Santa Clara, CA, USA, 29 October–1 November 2015; pp. 2823–2824.
42. Altché, F.; de La Fortelle, A. An LSTM network for highway trajectory prediction. In Proceedings of the 2017 IEEE 20th international conference on intelligent transportation systems (ITSC), Yokohama, Japan, 16–19 October 2017; pp. 353–359.
43. Xue, H.; Huynh, D.Q.; Reynolds, M. SS-LSTM: A hierarchical LSTM model for pedestrian trajectory prediction. In Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 12–15 March 2018; pp. 1186–1194.
44. Peng, C.; Cheng, Q. Discriminative Ridge Machine: A Classifier for High-Dimensional Data or Imbalanced Data. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 2595–2609. [[CrossRef](#)]
45. Scislo, L. High Activity Earthquake Swarm Event Monitoring and Impact Analysis on Underground High Energy Physics Research Facilities. *Energies* **2022**, *15*, 3705. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.