applied sciences

MDPI

Article

Predicting the Remaining Time before Earthquake Occurrence Based on Mel Spectrogram Features Extraction and Ensemble Learning

Bo Zhang [1], Tao Xu [1], Wen Chen [2,*] and Chongyang Zhang [1]

[1] School of Petroleum, China University of Petroleum-Beijing at Karamay, Karamay 834000, China; 2022216729@st.cupk.edu.cn (B.Z.); xutao@cupk.edu.cn (T.X.); zcy@st.cupk.edu.cn (C.Z.)
[2] School of Liberature and Science, China University of Petroleum-Beijing at Karamay, Karamay 834000, China
* Correspondence: chenwen@cupk.edu.cn

Abstract: Predicting the remaining time before the next earthquake based on seismic signals generated in a laboratory setting is a challenging research task that is of significant importance for earthquake hazard assessment. In this study, we employed a mel spectrogram and the mel frequency cepstral coefficient (MFCC) to extract relevant features from seismic signals. Furthermore, we proposed a deep learning model with a hierarchical structure. This model combines the characteristics of long short-term memory (LSTM), one-dimensional convolutional neural networks (1D-CNN), and two-dimensional convolutional neural networks (2D-CNN). Additionally, we applied a stacking model fusion strategy, combining gradient boosting trees with deep learning models to achieve optimal performance. We compared the performance of the aforementioned feature extraction methods and related models for earthquake prediction. The results revealed a significant improvement in predictive performance when the mel spectrogram and stacking were introduced. Additionally, we found that the combination of 1D-CNN and 2D-CNN has unique advantages in handling time-series problems.

Keywords: seismic signal; mel spectrograms; mel frequency cepstral coefficient; deep learning; stacking

Citation: Zhang, B.; Xu, T.; Chen, W.; Zhang, C. Predicting the Remaining Time before Earthquake Occurrence Based on Mel Spectrogram Features Extraction and Ensemble Learning. *Appl. Sci.* **2023**, *13*, 12268. https://doi.org/10.3390/app132212268

Academic Editors: Edoardo Rotigliano, José A. Peláez, Peidong Shi and Sanyi Yuan

Received: 12 October 2023
Revised: 29 October 2023
Accepted: 10 November 2023
Published: 13 November 2023

1. Introduction

Earthquakes are destructive natural disasters caused by the lateral sliding or intersection of tectonic plates. The occurrence of earthquakes is not a periodic process, as it depends on the nature of the area prone to rupture due to the partial release of stress and the differences in the types of various other faults [1]. This indicates that the periods between these earthquake events should be essentially irregular. The increasing integration between seismology, rock mechanics, and geology, as well as the significant impact of the plate tectonics concept on Earth sciences, were discussed by Hammond and Healy, J.H. [2,3]. These studies express optimism about earthquake prediction. Scholz, C. [4] analyzed crustal movements, tilting, fluid pressure, electric and magnetic fields, and radon emissions from the perspective of earthquake precursor effects, suggesting that these anomalies may be related to the occurrence of earthquakes. Previous earthquake prediction research was predominantly based on seismological theory and geological practical experience, but these methods often face challenges, such as low prediction accuracy and difficulties in precise location determination.

In recent years, machine learning, as a rapidly evolving field, has achieved significant success in various areas. Regression problems are common in everyday life, such as house price prediction, stock market forecasting, and predicting the remaining lifespan of machine parts. Chen, Y. et al. [5] used five popular machine learning and in-depth learning approaches to predict house prices and evaluated the models. Researchers such as Mehtab, S. et al. [6] have indicated that a univariate model based on LSTM, utilizing data from one week prior as input, provided the most accurate predictions for the opening prices of

NIFTY 50 time series in the following week. Additionally, Ma, N. et al. and Li, W. et al. [7,8] effectively employed a CNN and LSTM to predict the remaining lifespan of mechanical components under different environmental conditions. Machine learning is also widely applied in the field of earthquake prediction. For example, Laurenti, L. et al. [9] introduced an autoregressive (AR) model to predict the shear stress in fault zones; Asim, K. et al. and Cao, C. et al. [10,11] predicted earthquake magnitude using ML-based methods; Wang, K. et al. [12] utilized transfer learning to train a convolutional encoder-decoder to predict instantaneous and future fault sliding; Wang, X. et al. [13] used XGBoost, random forest (RF), and a deep neural network (DNN) to predict site amplification factors for different periods; Sang, K.H. et al. [14] proposed a novel ML-based reservoir prediction method based on virtual sample generation; Feng, T. et al. [15] showed that ML-based phase-picking exhibited good performance in constraining high-precision earthquake catalogs and constructing high-resolution velocity models; and Wrona, T. et al. [16] utilized ML for earthquake phase analysis to address the subjective issue of earthquake interpretation. Accurately predicting earthquakes is crucial to mitigating their societal impact.

In this study, the goal of predicting the remaining time before the next earthquake occurs based on seismic signals generated in a laboratory setting [17,18] was considered. Various machine learning (ML) algorithms were employed, including gradient boosting decision trees (GBDT), convolutional neural networks (CNN), long short-term memory (LSTM) networks, and model fusion (stacking). GBDT is an ensemble learning method [19], which builds a series of decision trees by boosting the mispredictions in the training samples, and makes final predictions by taking the weighted average of the predictions from all the decision trees. CNNs are deep learning algorithms that are particularly suited for processing image and signal data [20–22]. These networks use convolutional layers to extract features from the input data and pooling layers to reduce the dimensionality of the feature maps. LSTM is superior to recurrent neural networks (RNN) and is designed to address the long-term dependency problem in traditional RNN, and has demonstrated great potential for modeling time-series data [23–25]. Stacking is a powerful model ensembling strategy that can improve the prediction accuracy of machine learning models. Stacking is a very effective technique in various machine learning tasks, especially in cases where the best performance is desired [26–28]. In our work, we used the LightGBM model in GBDT and a deep learning model with a multi-level structure, finally feeding the prediction results from multiple models as new features into a meta-model, usually a linear model, such as linear regression, logistic regression, or support vector machine.

This study aims to contribute to the development of improved earthquake prediction models, which can help save lives and mitigate the societal and economic impacts of earthquakes. In this work, we combine mel frequency cepstral coefficients (MFCC) and the mel spectrograms of audio signals to extract features from denoised signals (see Section 2 for details). We introduce a two-dimensional convolutional neural network (CNN) to process the time-series data, enabling the simultaneous capture of local features in both the time and frequency domains. Furthermore, we apply a stacking model fusion strategy to achieve the best performance of the models. The remaining sections of the article are organized as follows: Section 2 introduces the data analysis and feature preprocessing; Section 3 elaborates on the design and implementation of the model structure. The experiments are described in Section 4. Section 5 summarizes the research achievements of this article and proposes further research directions.

2. Data Preprocessing

2.1. Datasets

The seismic signals were recorded by piezoelectric ceramic transducers located on the biaxial equipment side blocks. Each amplitude burst corresponds to a laboratory earthquake. We sought to predict the timing of the next laboratory earthquake based on a small subset of snapshots from the seismic data [23]. The dataset was divided into a training dataset and a testing dataset, with the training data accounting for 13% of the total data and

the test data accounting for 87%. The Train dataset consists of 16 CSV files, representing 16 instances of earthquakes occurring within a certain period of time. Each record in the dataset contains two fields: time-to-failure (TTF) and acoustic-data. All the training data and some test data are shown in Figure 1.

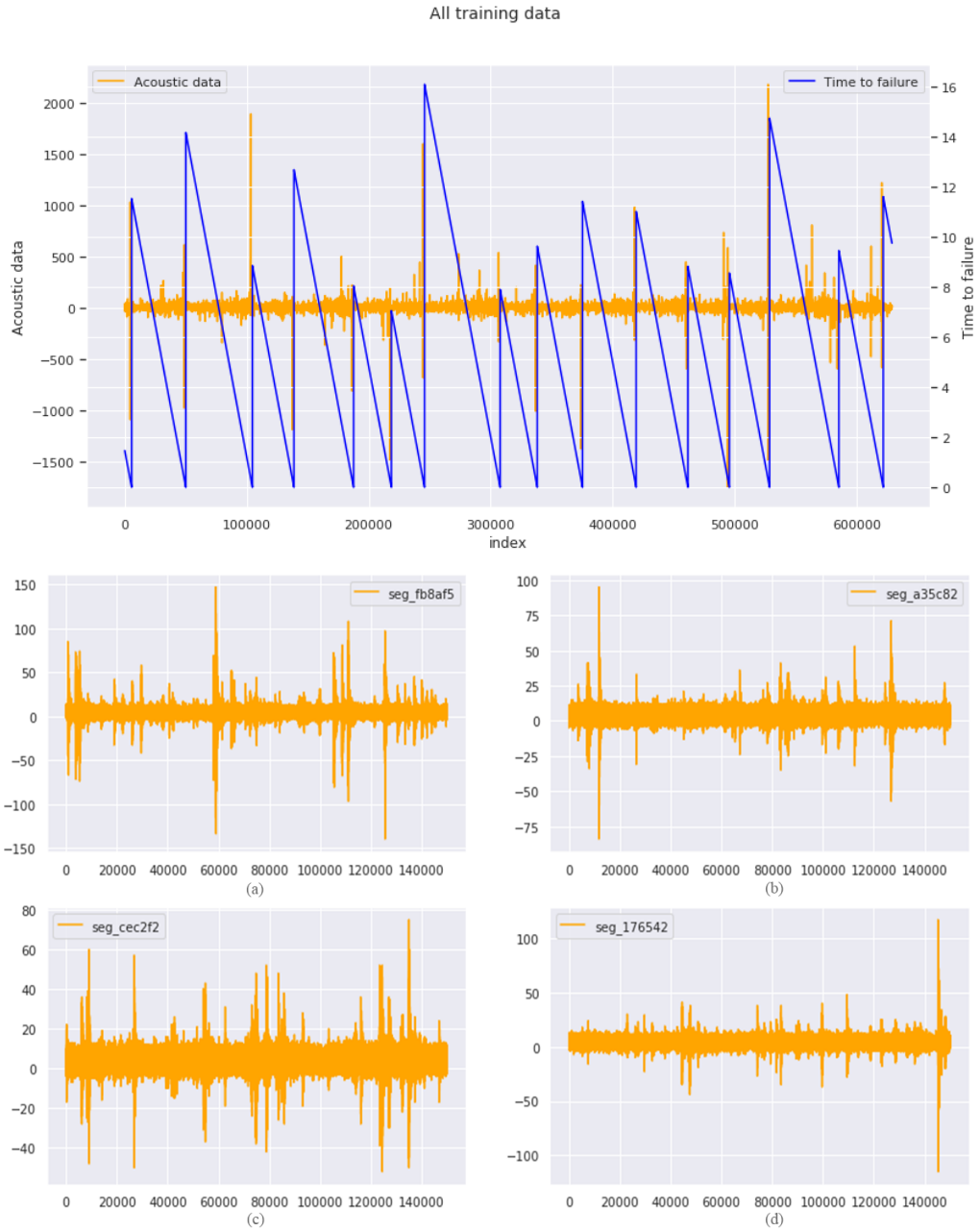

Figure 1. (**Top**): All training data illustration. The orange line is the acoustic data and the blue one is the TTF. The training data contain 629.145 million rows; it should be noted that due to the large amount of data, the step size of the graph is 1000. There are 16 earthquakes in the training data. The shortest time to failure is 1.5 s for the first earthquake, while the longest is around 16 s. (**a**–**d**) Partial test data illustration. Similarly, the orange line is the acoustic data. Each test seismic signal file has 150,000 rows.

2.2. Removing Noise

In data preprocessing, it is important to remove noise. For earthquake signals, the main problem is that the measured earthquake signals do not accurately represent the actual earthquake signals we are trying to discover. These pulse signals interact with the real earthquake signals (i.e., the signals we need) and generate the final mixed signal that is received by our seismometer (this process also occurs in earthquake simulation laboratories). Therefore, the main goal of seismology is to accurately identify and separate the real earthquake signals from the mixed signals. Wavelet denoising is a relatively general signal denoising method that can be applied to various types of signals, such as audio signals, image signals, biological signals, etc. [29–32]. Its advantage is that it can effectively remove noise in the signal while retaining the important features of the signal. It should be noted that the original signal we are dealing with is the convolution of artificial pulse signals and real earthquake signals, which is why we need to perform wavelet decomposition. The process is similar to deconvolution, which can reverse the convolution process and extract the real earthquake pulse signals from the mixed seismic records and artificial pulse signals [33–36]. To understand the randomness in the signal, we use the median absolute deviation (MAD) value to determine the minimum threshold of the wavelet coefficients in the time series [37,38]. Then, we filter out the low coefficients from the wavelet coefficients and reconstruct the real earthquake signals from the remaining coefficients. This successfully removes the pulse signals from the seismic records and obtains the real earthquake signals, as shown in Figure 2.

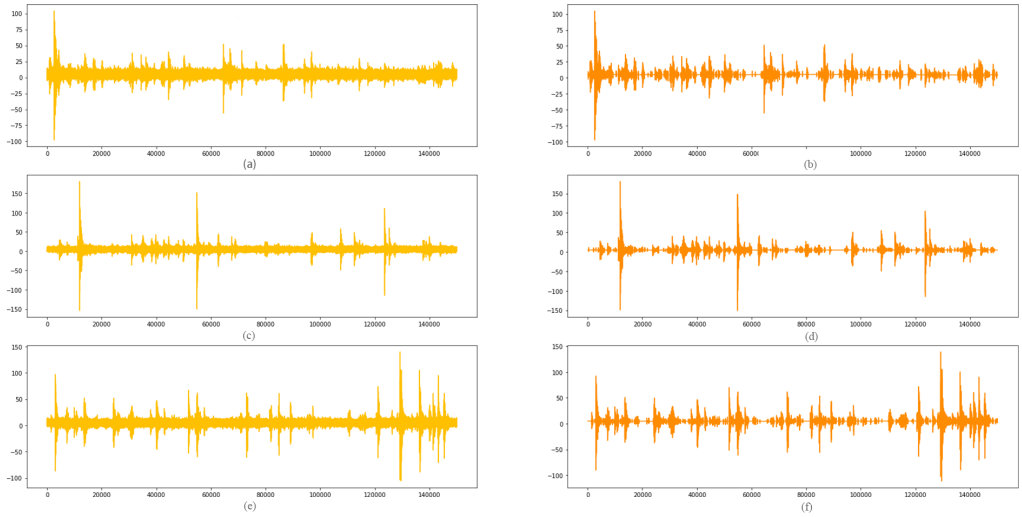

Figure 2. Comparison of denoised seismic signal data. The data used in this figure come from the test dataset, with the left column (**a**,**c**,**e**) showing the original signal in lighter colors and the right column (**b**,**d**,**f**) showing the signal data after wavelet denoising. The denoised seismic signal is smoother compared to the original signal, with significantly reduced noise amplitude and more prominent signal peaks. This makes the signal features more obvious and it is easier to observe the trend and characteristics of the signal.

2.3. Selecting Features

We applied audio feature extraction methods to this work, treating the seismic signal as audio. We combined the mel frequency cepstral coefficients (MFCCs) and the mel spectrogram of the audio signals to extract features from the denoised signal, and selected some of the most important features, as shown in Figure 3, for the model [39,40]. MFCC, which stands for mel frequency cepstral coefficient, is a feature representation method for audio signals mainly used in audio processing fields, such as audio signal classification, speech recognition, and music information retrieval [41–43]. The MFCC was originally designed to simulate the human perception of sound. Our ears are not equally sensitive to loudness—they are more sensitive to signals with higher frequencies than those with lower

frequencies. To simulate this non-linear frequency response, the MFCC uses a set of mel filters to simulate the frequency response characteristics of the human ear. Mel filters can filter audio signals in the frequency domain, allowing for more detailed characterization of frequency details in the low frequency range and coarser characterization of frequency details in the high frequency range. The calculation method of a mel spectrogram is as follows: preprocess the original audio signal, including removing DC bias, framing, and windowing; perform fast Fourier transform (FFT) on each time window to obtain the frequency spectrum of the signal; map the frequency spectrum to the mel scale, which is a non-linear frequency scale related to human auditory characteristics [44,45]—this process can be achieved by applying a linear transformation and a set of triangular filters; calculate the energy output of each triangular filter on the mel scale, which can weight the energy in different frequency ranges and merge them into one value—these energy values form one row of the mel spectrogram, with each row representing a time window; convert the values on the mel spectrogram from a linear scale to a logarithmic scale by taking the logarithm. Since the frequency range of seismic signals is wide, the mel frequency scale can more finely characterize the high-frequency part of the spectrum and compress the low-frequency part, thereby better exhibiting the characteristics of the seismic signals. Therefore, by observing the mel spectrogram, we can better understand the frequency spectrum characteristics of the seismic signals and be helped to classify and recognize the seismic signals, as shown in Figure 4.

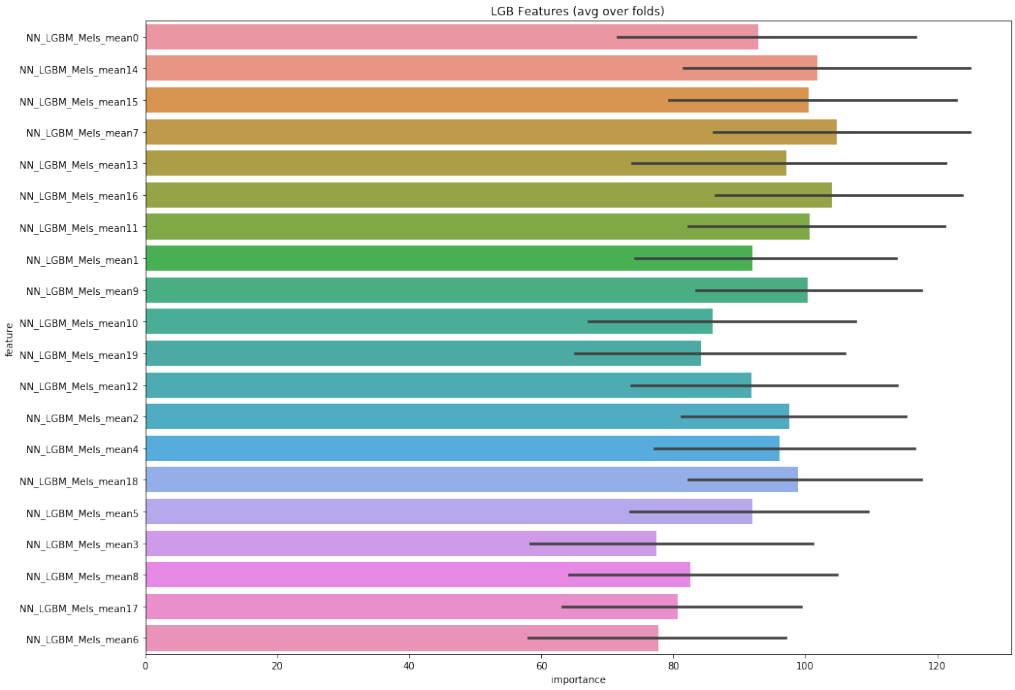

Figure 3. Comparison of the average value of the mel spectral coefficient on Lightgbm model importance. Adding several features with high importance to the model training greatly improved the score. It should be noted here that the selection of other features also applies this method to the extraction of features with high model importance, such as the number of peaks of at least support 2 on the denoised signal and above the 20% percentile on the std of the rolling window of size 50.

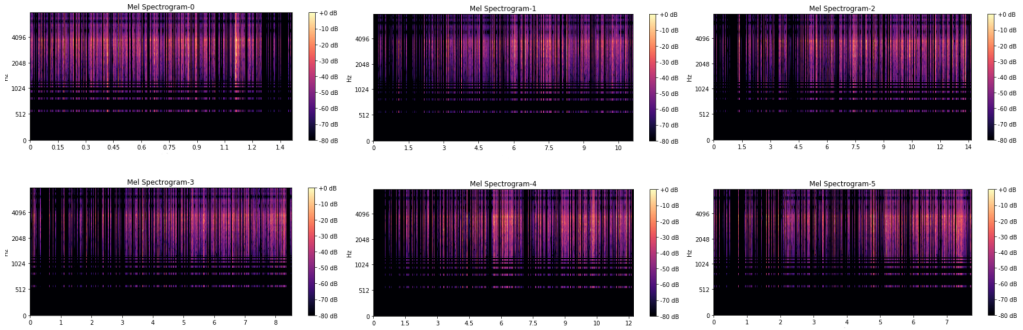

Figure 4. Mel spectrogram illustration. Please note that this graph only shows a part of the seismic data mel spectrogram, and it can be observed that the frequency of the seismic signal is concentrated between 2 MHz to 4 MHz. The highlighted portion in the graph represents the primary frequency or seismic signal, which is mixed with some noise components.

3. Earthquake Prediction Models

Our LightGBM model and the output layer of the neural network are consistent with the work of Levinson et al. We optimized the neural network structure on this basis, as shown in Figure 5. Finally, we combined the outputs of both models as the input for the next meta-model, performing model fusion.

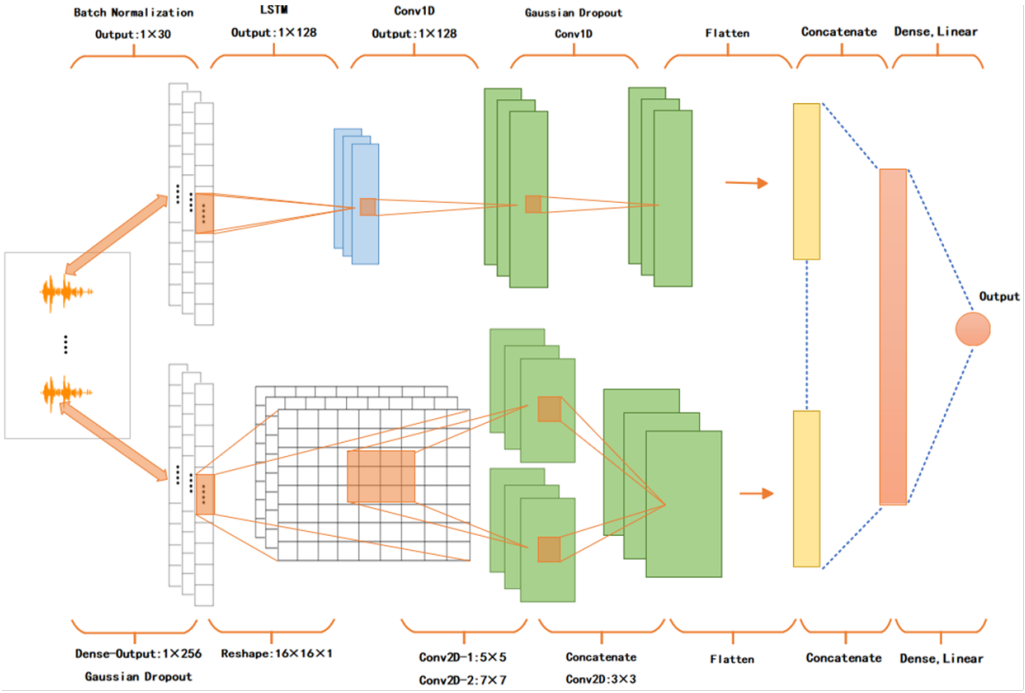

Figure 5. Schematic model of a deep learning model with a multi-level structure. We combined long short term memory (LSTM), one-dimensional convolutional neural networks (1D-CNN), and two-dimensional convolutional neural networks (2D-CNN). In the upper part of the diagram, the blue color represents the LSTM layer, followed by two 1D-CNN layers indicated by the green color, and the yellow color represents the flatten layer. In the lower part of the diagram, the input data dimension is modified to fit the 2D-CNN. Similarly, the green color represents the 2D-CNN, where the outputs of convolutional layers with different kernels are concatenated and inputted into the next 2D-CNN layer, and finally, the result is flattened. After processing the branches of the 1D-CNN and 2D-CNN, we concatenate their outputs using the concatenate operation to achieve feature fusion.

3.1. The Deep Neural Network Model

In this study, we designed and implemented a deep learning model with a multi-level structure, aiming to fully capture the temporal information and local features in the earthquake signal data. The model integrates the characteristics of long short-term memory (LSTM), 1D convolutional neural networks (1D-CNN), and 2D convolutional neural networks (2D-CNN). These structures collaborate with each other, jointly processing the input earthquake signal data to extract the temporal and local spatial features. The LSTM layer is responsible for processing the input time-series data. It effectively captures the long-term dependencies in the sequence data through memory cells and gate units. Following this, the one-dimensional convolutional neural network (1D-CNN) slides the convolutional kernels along the time dimension, enabling the detection of local features at different time scales. As the kernels slide across the entire sequence, this parameter sharing characteristic enhances the model's efficiency, reducing the number of trainable parameters. Subsequently, the data are input into a two-dimensional convolutional layer. While typically used for processing image data, in this model, we introduced the two-dimensional convolutional layer by appropriately transforming the data shape. By utilizing kernels of various sizes, the 2D-CNN can capture features across different ranges. Smaller kernels capture local features, whereas larger kernels can detect features over broader ranges. Finally, the model outputs three prediction results: time to failure (TTF), time since failure (TSF), and a binary classification result. This multi-level, multi-structure deep learning model design enables us to fully explore the potential information in the earthquake signal data, providing a more accurate and reliable basis for earthquake prediction. Here are some details for each layer:

- Input Layer. The input of the model is a tensor with the shape of (1, N), where N represents the feature dimension of the earthquake signal data. We introduced the mel spectrogram in Section 2.2 of the article, and here $N = 30$.
- Batch Normalization Layer. In the second layer of the model, we employed a batch normalization layer to normalize the input data. Batch normalization is a technique that improves the training performance of neural networks. It can accelerate mode convergence, enhance training stability, and reduce the sensitivity of the model to the initialization of weights. The basic idea of batch normalization is to standardize each feature on each mini-batch of data so that the mean of the data is 0 and the variance is 1. Specifically, by calculating the mean μ_B and variance ${\sigma^2}_B$ of the mini-batch, we can perform centering and standardization on the input data X_i, obtaining the normalized data. Subsequently, the normalized data are scaled and shifted using the learnable scale factor γ and shift factor β to obtain the final output result Y_i. This process helps to improve the training performance and stability of neural networks and reduce the reliance on parameter initialization. Batch normalization applies the following transformation to each feature i of the input data.

  Compute the mean of the mini-batch:

$$\mu_B = \frac{1}{m}\sum_{i=1}^{m} X_i \tag{1}$$

where m is the number of samples in the mini-batch, and X_i is the i-th sample.

Calculate the variance of the mini-batch:

$$\sigma_B^2 = \frac{1}{m}\sum_{i=1}^{m}(X_i - \mu_B)^2 \tag{2}$$

Normalize each feature:

$$\hat{X}_i = \frac{X_i - \mu_B}{\sqrt{{\sigma^2}_B + \varepsilon}} \tag{3}$$

where ε is a small positive number to prevent division by zero.

Perform scaling and shifting operations:

$$Y_i = \gamma^{\star} \hat{X}_i + \beta \tag{4}$$

where γ and β are learnable parameters that represent the scaling factor and shifting factor, respectively. These two parameters enable the model to have greater expressiveness and learn richer features from the input data.

- LSTM Layer. We employed a long short-term memory (LSTM) network to capture the long-term and short-term temporal dependencies in the earthquake signal data. LSTM is a special type of recurrent neural network (RNN) that addresses the vanishing and exploding gradient problems that traditional RNN may encounter when processing long sequences by using gating units. This makes LSTM more suitable for handling data with complex temporal dependencies, such as earthquake signals. In this model, we used an LSTM layer with 128 neurons to fully extract the temporal information in the input data. To further process the temporal features in subsequent layers, we set the return-sequences = True parameter, allowing the LSTM layer to output a sequence with the same length as the input. In this way, the LSTM layer can capture not only short-term fluctuations in the earthquake signal but also long-term trends in the earthquake sequence. By combining the LSTM layer and subsequent convolutional layers, our model can better capture and understand the dynamic characteristics of earthquake signal data, providing a more accurate and reliable basis for earthquake prediction.
- 1D-CNN Branch. After the LSTM layer, we constructed a subnetwork containing two 1D convolutional layers to extract local features from the earthquake signal. The first 1D convolutional layer has 128 convolution kernels, a kernel size of 2, and uses the ReLU activation function. The second 1D convolutional layer has 256 convolution kernels, a kernel size of 2, and also uses the ReLU activation function. Next, we employed a Gaussian dropout layer to prevent overfitting, and flattened the result for subsequent processing. Flattening is a data preprocessing operation in neural networks used to convert a multi-dimensional tensor into a one-dimensional tensor. This operation is very common in convolutional neural networks (CNN) because the feature maps usually have a multi-dimensional structure after multiple layers of convolution and pooling operations. In order to connect these feature maps to subsequent fully connected layers for further processing, we need to flatten them into a one-dimensional vector. Specifically, the operation of the flatten layer can be expressed with the following formula:

$$Y_i = x[a_1, a_2, \ldots, a_n] \tag{5}$$

In the formula, x is the input multi-dimensional tensor with a shape of $(d_1, \ldots, d_n)$; Y is the output one-dimensional tensor with a shape of m, where $m = d_1{}^{*} d_2{}^{*} \ldots {}^{*} d_n$; i is the element index in the output vector Y, with a range of $[0, m-1]$; $a_1, a_2, \ldots, a_n$ are the element indices in the input tensor x, with respective ranges of $[0, d_1 - 1]$, $[0, d_2 - 1]$, $\ldots$, $[0, d_n - 1]$; for each i. We can find a unique set of indices $a_1, a_2, \ldots, a_n$ such that Y_i corresponds to $x[a_1, a_2, \ldots, a_n]$. This process essentially involves traversing all elements of the input tensor x in a specific order (usually row-major or column-major) and copying them one-by-one into the output vector Y.

- 2D-CNN Branch. It is worth noting that after the input layer, we constructed a subnetwork consisting of a fully connected layer and multiple 2D convolutional layers. First, the data were reshaped into a tensor with a shape of (16, 16, 1). Next, we built two 2D convolutional layers with different kernel sizes, and then concatenated the outputs of the two 2D convolutional layers. Finally, the result was flattened for subsequent processing. 2D-CNN have certain advantages when processing time-series data, as they can capture local features in both the time and frequency domains simultaneously. Specifically, 2D-CNN can identify specific patterns at different times and frequen-

cies, revealing spatial (time-domain) and frequency features in the signal. This is significant for analyzing complex time-series data, such as earthquake signals. Using convolution kernels of different sizes can help the model capture features at different scales. Smaller kernels (e.g., 5×5) can capture local, detailed features in the signal, while larger kernels (e.g., 7×7) can capture a broader spatial range, thus identifying larger-scale patterns in the signal. This multi-scale feature extraction method helps improve the model's performance and generalization capabilities. Additionally, we used a technique called "dilated convolution" to increase the receptive field range. By increasing the dilation rate, the model can expand the receptive field range without increasing the kernel size. This helps capture more extensive contextual information while maintaining relatively low computational complexity. Finally, the feature maps after the convolution operation are concatenated through the concatenate layer, combining the different scale features captured by the two groups of convolutional layers. In summary, by using multi-scale convolution kernels and feature fusion strategies, the 2D-CNN branch can simultaneously capture rich features in the time and frequency domains, providing strong support for earthquake prediction tasks.

- Feature Concatenation Layer. After processing the outputs of the 1D-CNN and 2D-CNN branches, we concatenate them, achieving feature fusion. This design aims to combine the advantages of 1D-CNN and 2D-CNN to fully extract the features of the earthquake signal data.
- Dense Layer. After the feature fusion, we added a fully connected layer with 128 neurons and a ReLU activation function for further processing of the feature information.
- Outputs Layer. The model has three outputs, which are used to predict the time to failure (TTF), time since failure (TSF), and a binary classification result (to determine whether TTF is less than 0.5 s). The TTF output uses a fully connected layer with one neuron and a ReLU activation function. The TSF output also employs a fully connected layer with one neuron. For the binary classification output, a fully connected layer with one neuron and a sigmoid activation function is used.

3.2. Stacking

Stacking is a commonly used machine learning technique that aims to improve the prediction accuracy and model robustness by integrating the predictions of multiple models. In practical applications, different models may have varying abilities to recognize certain features or patterns in the data. By fusing the predictions of these models, we can leverage the strengths of each model and reduce the prediction errors caused by biases or noise in a single model. Model fusion can help us achieve better performance than a single model in many situations. In our work, we apply the stacking strategy to the time-series prediction task to enhance the prediction performance.

The specific steps are as follows:

Step 1: We first create a simple linear regression model (LR), with its expression as follows:

$$Y = X^{*}\beta + \varepsilon \tag{6}$$

where Y is the target variable (response variable) vector, X is the feature matrix (explanatory variable matrix), β is the model parameter (coefficient) vector, and ε is the error vector. Linear regression aims to find a set of parameters β that minimizes the sum of squared errors ε, minimizing the mean squared error (MSE). This method is called ordinary least squares (OLS).

Step 2: Stack the prediction results of the LightGBM model and the deep learning model together to serve as the input features for the second-layer model.

Step 3: Stack the test set prediction results of the LightGBM model and the deep learning model together to serve as the input features for testing the second-layer model.

Step 4: Train the linear regression model using the input features from Step 2 and the target values.

Step 5: Use the trained linear regression model to make predictions on the test set from Step 3, obtaining the combined test set prediction results.

4. Experiments

4.1. Model Training

In this section, to demonstrate the advantages of feature engineering and modeling, we conducted two sets of experiments. We used 3-fold cross-validation to train the model based on the available data. We applied different models to two sets of feature data (Dataset 1 and Dataset 2), using the mean absolute error (MAE) as the final evaluation metric. Each model was trained for 1000 epochs, and the batch-size was 256. The formula is as follows:

$$MAE = \frac{1}{m}\sum_{i=1}^{m}|X_i - Y_i| \tag{7}$$

where m is the number of samples, X_i is the true value, and Y_i is the predicted value.

In Experiment 1, Dataset 1 represents the feature dataset used for training by (https://www.kaggle.com/competitions/LANL-Earthquake-Prediction/discussion/94390 (accessed on 11 October 2023)), and in Experiment 2, Dataset 2 is based on this and adds some new features (Table 1).

Table 1. New features added in this study.

Index	Feather Name	Feather Description
1	NN_Mels_mean4	original signal in the 5th Mel frequency band
2	NN_Mels_mean5	original signal in the 6th Mel frequency band
3	NN_Mels_mean18	original signal in the 19th Mel frequency band
4	NN_LGBM_Mels_mean9	original signal in the 10th Mel frequency band
5	NN_LGBM_Mels_mean11	original signal in the 12th Mel frequency band
6	NN_LGBM_Mels_mean16	riginal signal in the 17th Mel frequency band
7	NN_Mels_mean_denoise_6	denoised signal in the 7th Mel frequency band
8	NN_Mels_mean_denoise_7	denoised signal in the 8th Mel frequency band
9	NN_Mels_mean_denoise_13	denoised signal in the 14th Mel frequency band

4.2. Experimental Results of Model Performance

We selected four model structures, which were: a single LightGBM (LGB) model; a model stacking LSTM and 1D-CNN (NN-1D); a model stacking LSTM, 1D-CNN, and 2D-CNN (NN-2D); and finally, using the prediction results of LGB and NN-2D as input, employing a linear regression model as the meta-model for model fusion (stacking). Train-MAE and Test-MAE correspond to the scores of the training data and the testing data, respectively. The final results are based on the testing data.

Evaluating and comparing the performance of models on the training and validation datasets is crucial. By observing the MAE curves of different models, we can quickly identify which models perform better for specific tasks. Moreover, if the training MAE continues to decrease, but the validation MAE begins to rise after reaching the minimum value, it is usually an indication of overfitting in the model. Here, the MAE comparisons of different neural network structures in the two sets of experiments separately are shown in Figures 6 and 7.

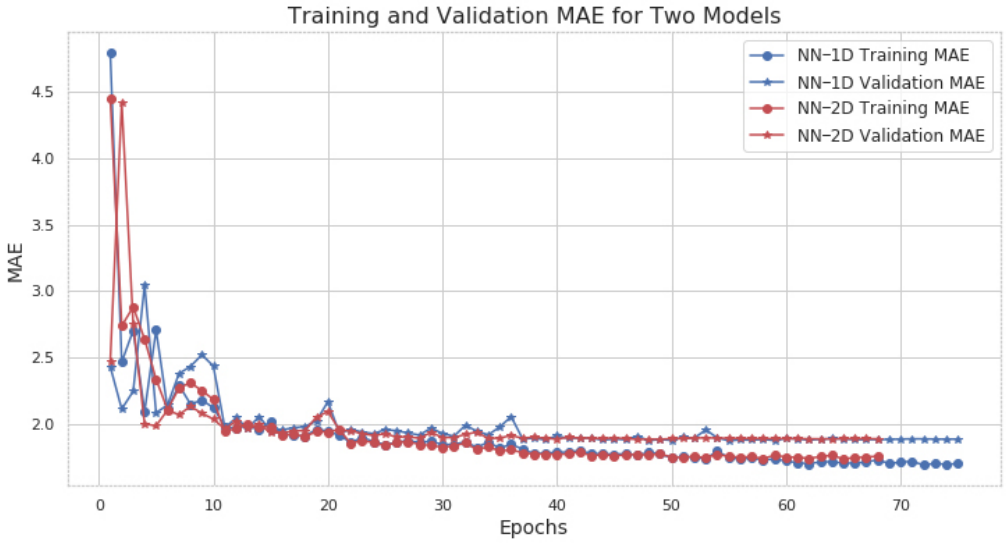

Figure 6. For the first experiment, the training and validation MAE for the two neural network structures (NN-1D and NN-2D) are shown. Solid lines marked with blue circles and star-shaped markers represent the training MAE and the validation MAE of the NN-1D model, respectively, while solid lines marked with red circles and star-shaped markers represent the training MAE and validation MAE of the NN-2D model, respectively. It can be clearly seen that the performance of the two models is very similar in this dataset, and there are no overfitting issues for either model.

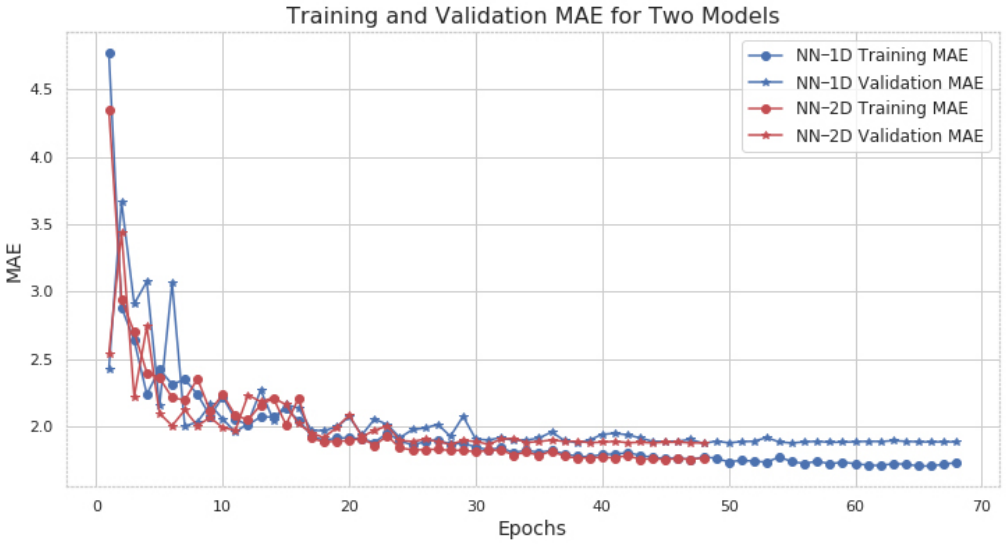

Figure 7. In the second experiment, the training and validation MAE for the two neural network structures (NN-1D and NN-2D) are shown. The information represented by the four curves in the figure is the same as in Figure 8. It is found that NN-2D converges faster to fitting on the validation set compared to NN-1D.

The experimental results show that the performance of the NN-2D neural network was better than that of the NN-1D under the same dataset. As shown in Tables 2 and 3, we can see that the MAE of NN-2D was higher than that of NN-1D during the training process, but NN-2D achieved better results in the testing process. This is because in our work, we used 13% of the data as training and validation data and 87% as test data, so there is some overfitting in the final results. This proves that the NN-2D model has better generalizability. Comparing the same model in both tables shows that the model scores under Dataset 2 are better than Dataset 1. This result demonstrates that the mel spectrogram is beneficial for extracting effective features from earthquake signals. As shown in Table 3, our best

result was 2.23831; the stacking strategy can more accurately predict the remaining time before the next laboratory earthquake. For the results of other participants, please refer to Johnson, P.A. et al. [23]. The prediction results are shown in Figure 8.

Table 2. Comparison of the scores of each model in Dataset 1.

Model	Train-MAE	Test-MAE
LGB	1.89379	2.27931
NN-1D	1.86133	2.28126
NN-2D	1.86481	2.27835
Stacking	1.85483	2.24514

Table 3. Comparison of the scores of each model in Dataset 2.

Model	Train-MAE	Test-MAE
LGB	1.89362	2.27851
NN-1D	1.85528	2.27628
NN-2D	1.86463	2.27388
Stacking	1.85374	2.23831

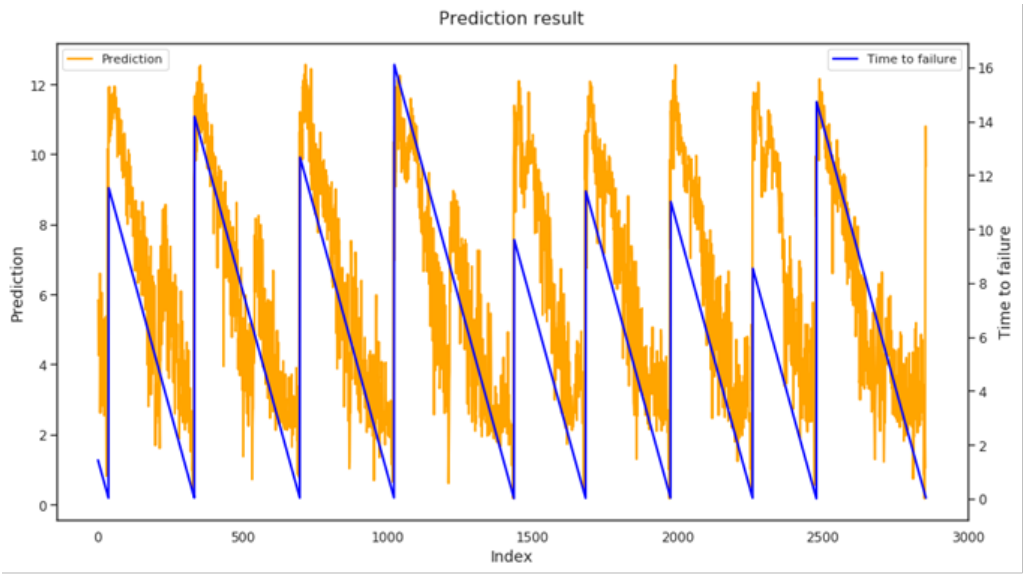

Figure 8. Comparison of predictions and actual values. The blue line represents the remaining time before the next laboratory earthquake occurs (actual values), and the orange line represents the model's predicted values.

5. Conclusions

In this paper, we focused on feature engineering and the performance of prediction models, achieving good results in the private leaderboard. The experiments demonstrated that mel spectrogram features, as a feature representation method for audio signals, can effectively capture the frequency characteristics of the signals, thereby improving the prediction accuracy. At the same time, we used 2D convolutional neural networks with different kernel sizes and dilated convolutions to predict time series, which can capture more effective features at different scales compared to a single 1D convolutional neural network. In addition, the stacking strategy further improved our prediction performance because different models have different recognition capabilities for the data. By leveraging the "wisdom" of multiple models, we can enhance the overall prediction performance and reduce the risk of overfitting for individual models. In summary, we successfully improved the performance of the prediction task by optimizing the feature engineering, the model

structure, and the model fusion. These optimizations offer valuable insights for handling similar problems in practical applications. However, accurately predicting earthquakes remains a challenging task. Laboratory earthquake data, while valuable, are a relatively idealized form of data and cannot fully simulate real earthquakes. The complex and variable nature of fault zones constitutes the most significant factor affecting the accuracy of models. In future work, we will continue to explore more feature extraction methods, more efficient model structures, and fusion strategies to achieve even better prediction performance.

Author Contributions: Conceptualization, W.C.; methodology, T.X.; software, C.Z.; writing—original draft preparation, B.Z.; writing—review and editing, B.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Basic Scientific Research Business Expenses for Universities in the Autonomous Region: Scientific Research Projects Grant Number XJEDU2023P171.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available from the corresponding author. Publicly available datasets were analyzed in this study. This data can be found here: https://www.kaggle.com/competitions/LANL-Earthquake-Prediction/discussion/94390 (accessed on 11 October 2023).

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

References

1. Aguirre, L.; Bataille, K.; Novoa, C.; Peña, C.; Vera, F. Kinematics of subduction processes during the earthquake cycle in Central Chile. *Seismol. Res. Lett.* **2019**, *90*, 1779–1791. [CrossRef]
2. Hammond, A.L. Earthquake prediction and control. *Science* **1971**, *173*, 316. [CrossRef]
3. Healy, J.H. Recent highlights and future trends in research on earthquake prediction and control. *Rev. Geophys.* **1975**, *13*, 361–364. [CrossRef]
4. Scholz, C. A physical interpretation of the Haicheng earthquake prediction. *Nature* **1977**, *267*, 121–124. [CrossRef] [PubMed]
5. Chen, Y.; Xue, R.; Zhang, Y. House price prediction based on machine learning and deep learning methods. In Proceedings of the 2021 International Conference on Electronic Information Engineering and Computer Science (EIECS), Changchun, China, 23–26 September 2021; pp. 699–702. [CrossRef]
6. Mehtab, S.; Sen, J.; Dutta, A. Stock Price Prediction Using Machine Learning and LSTM-Based Deep Learning Models. *arXiv* **2020**, arXiv:abs/2009.10819. [CrossRef]
7. Ma, N.; Yin, H.; Wang, K. Prediction of the Remaining Useful Life of Supercapacitors at Different Temperatures Based on Improved Long Short-Term Memory. *Energies* **2023**, *16*, 5240. [CrossRef]
8. Li, W.; Zhang, L.C.; Wu, C.H.; Wang, Y.; Cui, Z.; Niu, C. A data-driven approach to RUL prediction of tools. *Adv. Manuf.* **2023**, 1–13. [CrossRef]
9. Laurenti, L.; Tinti, E.; Galasso, F.; Franco, L.; Marone, C. Deep learning for laboratory earthquake prediction and autoregressive forecasting of fault zone stress. *Earth Planet. Sci. Lett.* **2022**, *598*, 117825. [CrossRef]
10. Asim, K.; Martínez-Álvarez, F.; Basit, A.; Iqbal, T. Earthquake magnitude prediction in Hindukush region using machine learning techniques. *Nat. Hazards* **2017**, *85*, 471–486. [CrossRef]
11. Cao, C.; Wu, X.; Yang, L.; Zhang, Q.; Wang, X.; Yuen, D.A.; Luo, G. Long short-term memory networks for pattern recognition of synthetical complete earthquake catalog. *Sustainability* **2021**, *13*, 4905. [CrossRef]
12. Wang, K.; Johnson, C.W.; Bennett, K.C.; Johnson, P.A. Predicting fault slip via transfer learning. *Nat. Commun.* **2021**, *12*, 7319. [CrossRef] [PubMed]
13. Wang, X.; Wang, Z.; Wang, J.; Miao, P.; Dang, H.; Li, Z. Machine learning based ground motion site amplification prediction. *Front. Earth Sci.* **2023**, *11*, 1053085. [CrossRef]
14. Sang, K.H.; Yin, X.Y.; Zhang, F.C. Machine learning seismic reservoir prediction method based on virtual sample generation. *Pet. Sci.* **2021**, *18*, 1662–1674. [CrossRef]
15. Feng, T.; Zhang, M.; Xu, L.; Wu, J.; Fang, L. Machine learning-based earthquake catalog and tomography characterize the middle-northern section of the Xiaojiang fault zone. *Seismol. Soc. Am.* **2022**, *93*, 2484–2497. [CrossRef]

16. Wrona, T.; Pan, I.; Gawthorpe, R.L.; Fossen, H. Seismic facies analysis using machine learning. *Geophysics* **2018**, *83*, O83–O95. [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. Hulbert, C.; Rouet-Leduc, B.; Johnson, P.A.; Ren, C.X.; Rivière, J.; Bolton, D.C.; Marone, C. Similarity of fast and slow earthquakes illuminated by machine learning. *Nat. Geosci.* **2019**, *12*, 69–74. [CrossRef]
19. Zhang, W.; Yu, J.; Zhao, A.; Zhou, X. Predictive model of cooling load for ice storage air-conditioning system by using GBDT. *Energy Rep.* **2021**, *7*, 1588–1597. [CrossRef]
20. Aymerich, E.; Sias, G.; Pisano, F.; Cannas, B.; Fanni, A.; the-JET-Contributors. CNN disruption predictor at JET: Early versus late data fusion approach. *Fusion Eng. Des.* **2023**, *193*, 113668. [CrossRef]
21. Hu, R.; Xiang, S. CNN prediction based reversible data hiding. *IEEE Signal Process. Lett.* **2021**, *28*, 464–468. [CrossRef]
22. Allam, J.P.; Samantray, S.; Sahoo, S.P.; Ari, S. A deformable CNN architecture for predicting clinical acceptability of ECG signal. *Biocybern. Biomed. Eng.* **2023**, *43*, 335–351. [CrossRef]
23. Johnson, P.A.; Rouet-Leduc, B.; Pyrak-Nolte, L.J.; Beroza, G.C.; Marone, C.J.; Hulbert, C.; Howard, A.; Singer, P.; Gordeev, D.; Karaflos, D.; et al. Laboratory earthquake forecasting: A machine learning competition. *Proc. Natl. Acad. Sci. USA* **2021**, *118*, e2011362118. [CrossRef]
24. Cao, J.; Li, Z.; Li, J. Financial time series forecasting model based on CEEMDAN and LSTM. *Phys. A Stat. Mech. Appl.* **2019**, *519*, 127–139. [CrossRef]
25. Sagheer, A.; Kotb, M. Time series forecasting of petroleum production using deep LSTM recurrent networks. *Neurocomputing* **2019**, *323*, 203–213. [CrossRef]
26. Taghizadeh-Mehrjardi, R.; Schmidt, K.; Amirian-Chakan, A.; Rentschler, T.; Zeraatpisheh, M.; Sarmadian, F.; Valavi, R.; Davatgar, N.; Behrens, T.; Scholten, T. Improving the spatial prediction of soil organic carbon content in two contrasting climatic regions by stacking machine learning models and rescanning covariate space. *Remote Sens.* **2020**, *12*, 1095. [CrossRef]
27. Shaw, R.; Lokshin, A.E.; Miller, M.C.; Messerlian-Lambert, G.; Moore, R.G. Stacking machine learning algorithms for biomarker-based preoperative diagnosis of a pelvic mass. *Cancers* **2022**, *14*, 1291. [CrossRef]
28. Chatzimparmpas, A.; Martins, R.M.; Kucher, K.; Kerren, A. StackGenVis: Alignment of data, algorithms, and models for stacking ensemble learning using performance metrics. *IEEE Trans. Vis. Comput. Graph.* **2020**, *27*, 1547–1557. [CrossRef]
29. Tian, C.; Zheng, M.; Zuo, W.; Zhang, B.; Zhang, Y.; Zhang, D. Multi-stage image denoising with the wavelet transform. *Pattern Recognit.* **2023**, *134*, 109050. [CrossRef]
30. Baldazzi, G.; Sulas, E.; Urru, M.; Tumbarello, R.; Raffo, L.; Pani, D. Wavelet denoising as a post-processing enhancement method for non-invasive foetal electrocardiography. *Comput. Methods Programs Biomed.* **2020**, *195*, 105558. [CrossRef]
31. Wang, L.; Zheng, W.; Ma, X.; Lin, S. Denoising speech based on deep learning and wavelet decomposition. *Sci. Program.* **2021**, *2021*, 8677043. [CrossRef]
32. Renfei, T.; Tao, L.; Min, O. Method for wavelet denoising of multi-angle prestack seismic data. *Acta Geophys.* **2023**, *71*, 1515–1524. [CrossRef]
33. Sui, Y.; Ma, J. A nonstationary sparse spike deconvolution with anelastic attenuation. *Geophysics* **2019**, *84*, R221–R234. [CrossRef]
34. Gao, Z.; Hu, S.; Li, C.; Chen, H.; Jiang, X.; Pan, Z.; Gao, J.; Xu, Z. A deep-learning-based generalized convolutional model for seismic data and its application in seismic deconvolution. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–17. [CrossRef]
35. Mirel, M.; Cohen, I. Multichannel Semi-blind Deconvolution (MSBD) of seismic signals. *Signal Process.* **2017**, *135*, 253–262. [CrossRef]
36. Ravasi, M.; Selvan, T.; Luiken, N. Stochastic multi-dimensional deconvolution. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–14. [CrossRef]
37. Li, Y.; Mei, L.; Li, R.; Wu, C. Using noise level to detect frame repetition forgery in video frame rate up-conversion. *Future Internet* **2018**, *10*, 84. [CrossRef]
38. Sahu, S.; Singh, H.V.; Kumar, B.; Singh, A.K. De-noising of ultrasound image using Bayesian approached heavy-tailed Cauchy distribution. *Multimed. Tools Appl.* **2019**, *78*, 4089–4106. [CrossRef]
39. Xie, T.; Zheng, X.; Zhang, Y. Seismic facies analysis based on speech recognition feature parameters. *Geophysics* **2017**, *82*, O23–O35. [CrossRef]
40. Zhou, Q.; Tong, G.; Xie, D.; Li, B.; Yuan, X. A seismic-based feature extraction algorithm for robust ground target classification. *IEEE Signal Process. Lett.* **2012**, *19*, 639–642. [CrossRef]
41. Alghamdi, N.S.; Zakariah, M.; Hoang, V.T.; Elahi, M.M. Neurogenerative Disease Diagnosis in Cepstral Domain Using MFCC with Deep Learning. *Comput. Math. Methods Med.* **2022**, *2022*, 4364186. [CrossRef]
42. Noda, J.J.; Travieso-González, C.M.; Sanchez-Rodriguez, D.; Alonso-Hernández, J.B. Acoustic classification of singing insects based on MFCC/LFCC fusion. *Appl. Sci.* **2019**, *9*, 4097. [CrossRef]
43. Fahad, M.S.; Deepak, A.; Pradhan, G.; Yadav, J. DNN-HMM-based speaker-adaptive emotion recognition using MFCC and epoch-based features. *Circuits Syst. Signal Process.* **2021**, *40*, 466–489. [CrossRef]

44. Borwankar, S.; Verma, J.P.; Jain, R.; Nayyar, A. Improvise approach for respiratory pathologies classification with multilayer convolutional neural networks. *Multimed. Tools Appl.* **2022**, *81*, 39185–39205. [CrossRef] [PubMed]
45. Coelho, G.; Matos, L.M.; Pereira, P.J.; Ferreira, A.; Pilastri, A.; Cortez, P. Deep autoencoders for acoustic anomaly detection: Experiments with working machine and in-vehicle audio. *Neural Comput. Appl.* **2022**, *34*, 19485–19499. [CrossRef]