

Article

Research on Automatic Classification of Coal Mine Microseismic Events Based on Data Enhancement and FCN-LSTM Network

Guojun Shang ^{1,2}, Li Li ², Liping Zhang ³, Xiaofei Liu ¹ , Dexing Li ^{1,2} , Gan Qin ² and Hao Li ^{4,*}

¹ School of Safety Engineering, China University of Mining and Technology, Xuzhou 221116, China; shangguojun@live.com (G.S.)

² Shenzhen Urban Public Safety Technology Research Institute Co., Ltd., Shenzhen 518000, China

³ Dagang Oilfield Foreign Cooperation Projects Department of PetroChina, Tianjin 300280, China

⁴ Yangtze Delta Region Institute (Huzhou), University of Electronic Science and Technology of China, Huzhou 313001, China

* Correspondence: lihao_geo@163.com

Abstract: Efficient and accurate classification of the microseismic data obtained in coal mine production is of great significance for the guidance of coal mine production safety, disaster prevention and early warning. In the early stage, the classification of microseismic events relies on human experiences, which is not only inefficient but also often causes some misclassifications. In recent years, the neural network-based classification method has become more favored by people because of its advantages in modeling procedures. A microseismic signal is a kind of time-series signal and the application of the classification method is widely optimistic. The number and the balance of the training data samples have an important impact on the accuracy of the classification result. However, the quality of the training data set obtained from the production cannot be guaranteed. A long short-term memory (LSTM) network can analyze the time-series input data, where the image classification at the pixel level can be achieved by the fully convolutional network (FCN). The two structures in the network can not only use the advantages of the FCN for extracting signal details but also use the characteristics of LSTM for conveying and expressing the long time-series information effectively. In this paper, a time-series data enhancement combination process is proposed for the actual poor microseismic data. A hybrid FCN-LSTM network structure was built, the optimal network parameters were obtained by experiments, and finally a reasonable microseismic data classifier was obtained.

Keywords: deep learning; microseismic data classification; data enhancement; FCN-LSTM network



Citation: Shang, G.; Li, L.; Zhang, L.; Liu, X.; Li, D.; Qin, G.; Li, H. Research on Automatic Classification of Coal Mine Microseismic Events Based on Data Enhancement and FCN-LSTM Network. *Appl. Sci.* **2023**, *13*, 11158. <https://doi.org/10.3390/app132011158>

Academic Editors: Yaojun Wang and Bangyu Wu

Received: 19 September 2023

Revised: 3 October 2023

Accepted: 8 October 2023

Published: 11 October 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

Hydrofracturing microseismic monitoring is a technique which images the volume of rock stimulated by hydraulic fracturing [1–3]. This technique also has been extensively applied to oil and gas reservoir dynamic monitoring since the 1960s [4–7]. In the microseismic monitoring process, except for the microseismic events generated by reservoir fracturing, the microseismic events generated by the activation of primary fractures may also be recorded [8–10]. The microseismic monitoring technique also plays a particularly important role in coal mines [11–15], in which uses the signals emitted in the coal or rock fracture process to proceed the mechanical stability analysis of coal or rock material and the associated rock structure. This technique is a real-time, dynamic, and continuous geophysics method [16–19].

Accurate and quick microseismic data classification in coal mine production is of great significance for the guidance of the safety and early warning of coal mine production. In recent years, classification methods based on machine learning have become popular in

modeling procedures. Long short-term memory (LSTM) is a special type of recurrent neural network (RNN), which can be used to analyze the time-series input data. LSTM can obtain the long-term dependence information and solve the long-term dependence problem in neural networks [20,21]. It can deal with the gradient vanishing or gradient explosion problems [22]. The way to overcome gradient exploding is gradient clipping, meaning that when the gradient exceeds threshold c or is less than threshold $-c$, the gradient is set as c or $-c$, where c is a constant. It is widely used in natural language processing (NLP) and temporal signal prediction and classification [23–25]. Therefore, the microseismic signal is a kind of temporal signal, and LSTM has great potential in microseismic signal classification.

A fully convolutional network (FCN) achieves image classifications at pixel level, solving the semantic-level image segmentation problem. Different from the classic CNN, the FCN can use input images of any size. It can upsample the feature map of the last volume layer, and then restore it to the same size as the input image through a deconvolution layer. Thus, a prediction can be generated for each pixel, while retaining the spatial information of the original image. Finally, parity is used to classify pixels in the upsampled feature map.

A successful neural network requires many parameters, which are obtained by many training data sets in the training process. The training process constantly iterates the training model and finally achieves a required neural network [26]. As a common data expansion technique, data enhancement is crucial for deep learning training models [27], which can effectively avoid model overfitting caused by small training sets [28].

Researchers have already noticed that the unbalanced distribution of data samples in the training data set will adversely affect the trained classifier [29]. These events ignored by the neural network lead to an incorrect classification result [30]. Many scholars have proposed some data enhancement methods, such as adding a random linear bend in the frequency domain, which enhances sound timing signals to achieve sound signal recognition [31]. Data sparsification without changing labels can be used for modeling of deep neural networks [32]. Reducing the original signal speed, adding noise, time and space/rotation distortions, and transformation of the feature domain can also realize the data enhancement [33–36].

Aiming at the collected small microseismic data sets and the unbalanced distribution of samples, this paper proposes a series of combination methods to achieve data enhancement and sample distribution balance to obtain the usable training data sets. The application of an FCN-LSTM network for classification in microseismic signals is also proposed in this paper. The reasonable parameters were obtained with multiple network training experiments, and the accuracies of the testing data set and validation data set are already over 90%. It is proved that the FCN-LSTM network is suitable. Based on the above, the proposed method can obtain a highly available network model under small and uneven distributed microseismic data sets and complete the real-time classification of microseismic events with high accuracy.

2. Data Enhancement

The original training set data in this paper comes from the recorded field microseismic waveform in the tunneling process of a working face in Xingdong Mine of Jizhong Energy Group from October 2018 to April 2019. After identification and classification, the data set contains three classes of microseismic signals, the typical waveforms of each class are demonstrated in Figure 1, and the numbers of each class are 2319, 361, and 314.

The three classes of events are groundwater movement, continuous rupture of coal seam, and tectonic activation-induced microseismic events.

It is observed that there are two problems when the data set is used as the training set data. On the one hand, the total number of training set samples is relatively small. On the other hand, the distribution is very unbalanced: the number of samples of the first class of data is large, and the number of samples of the other two types of data is relatively small, so the ratio of the three classes is close to 7:1:1. It is very difficult to obtain a classifier with

high accuracy using these training data. The role of data enhancement is mainly reflected in two aspects:

1. Increase the amount of training data to improve the generalization ability of the model.
2. Increase the noise data to improve the robustness of the model.

We designed the following process to achieve the purpose of data augmentation:

1. Bandpass filtering.
2. Upsampling.
3. Window warping and window slicing.

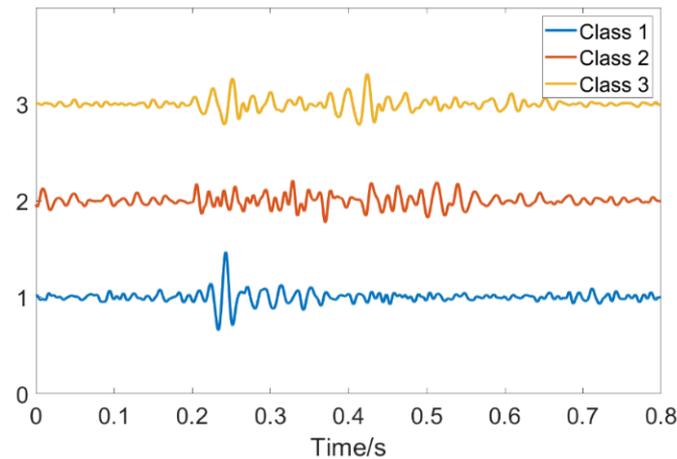


Figure 1. Schematic of three typical waveforms in the original training set.

2.1. Bandpass Filter

The dominant frequency distribution of the field data is shown in Figure 2. The dominant frequency of the effective signal is below 180 Hz. The purpose of the bandpass filter is to increase the number of training sets and prepare for upsampling. Figure 3 illustrates the waveform and spectrum of an example sample, whose label is 3, which has a time sampling interval of 0.2 ms and a length of 4000 sampling points.

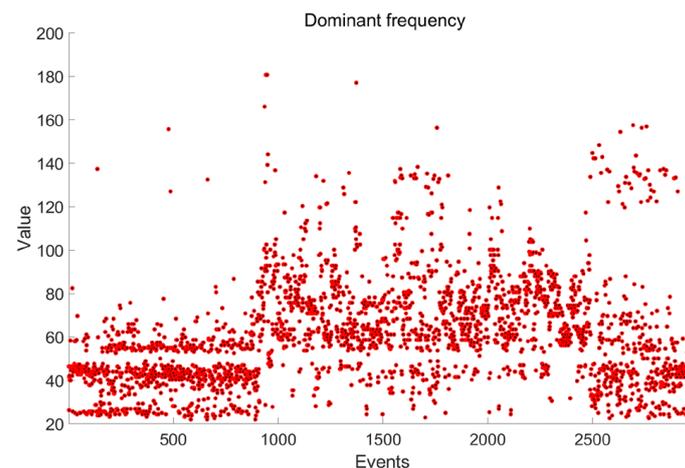


Figure 2. The dominant frequency distribution of original data set.

According to the spectrogram, the original waveform recording contains a lot of high-frequency noise. Bandpass filtering is used to improve the SNR and realize the purpose of increasing the number of training set samples. Figure 4 illustrates the waveform and spectrum after bandpass filtering. The comparison of Figures 3 and 4 illustrates that bandpass filtering does not change the label of the waveform.

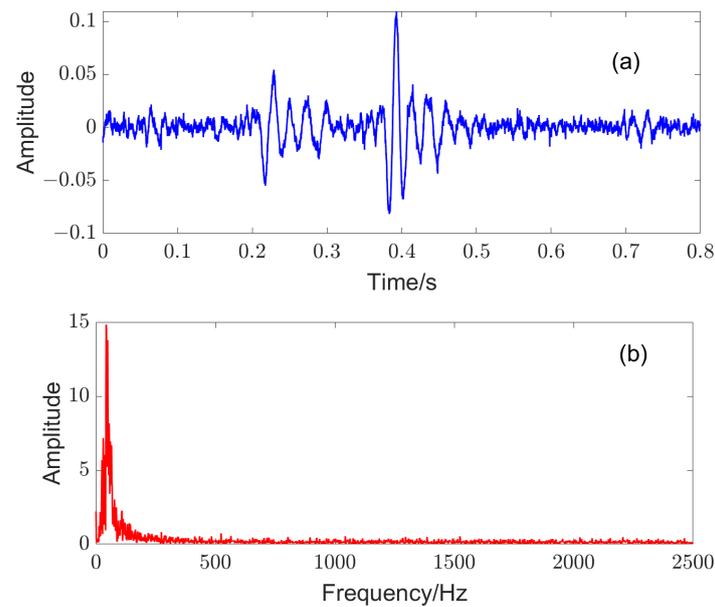


Figure 3. Waveform and spectrum of the original signal. (a) is the waveform of original signal, (b) is the spectrum of original signal.

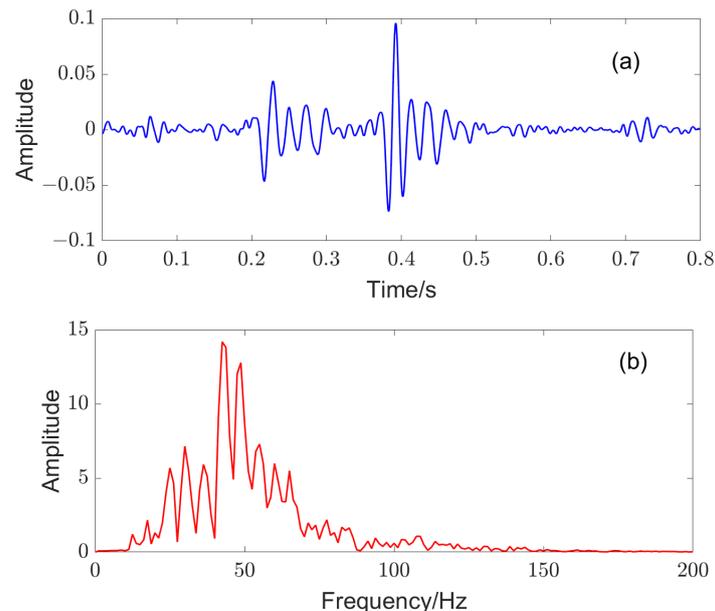


Figure 4. Waveform and spectrum after bandpass filtering. (a) is the waveform of signal after bandpass filtering, (b) is the spectrum of signal after bandpass filtering.

After bandpass filtering all the training data once, the number of samples is doubled:

$$N_{n1} = 2 \times N_0 = 2 \times 2994 = 5988 \quad (1)$$

where N_0 and N_{n1} are the total number of samples in the training data set before and after bandpass filtering, respectively. The number of samples of each class is doubled, but the ratio of the three classes is still close to 7:1:1, and the balance is not improved.

2.2. Upsampling

Sampling is a method that transforms the training set from an imbalanced data set to a balanced data set. The class with more samples in the data set is called the “popular class”, and the class with fewer samples is called the “minority class”. In the original training set

used in this paper, the first class of data can be called the “popular class”, and the other two types of data can be called the “minority class”.

The operation of making multiple copies of the minority class is called upsampling, and the operation of taking a partial sample from the popular class is called downsampling. Both sampling methods have their own shortcomings: some samples will appear repeatedly in the data after upsampling, and the trained model will have a certain over-fitting. Downsampling, on the other hand, loses some data, and the resulting model is only a part of the overall model. To avoid duplicate samples during upsampling, we can add some slight perturbations (random noise) to the newly generated samples to improve the balance of the data set without overfitting.

The next step is to verify the effect of adding noise on the sample labels. The bandpass-filtered signal is regarded as the effective signal, and the residual between the original signal and the effective signal is regarded as the noise. The *SNR* of the sample can be calculated by using the following equation.

$$SNR = 10 \lg \frac{\sum_{i=1}^N S(i)^2}{\sum_{i=1}^N N(i)^2} \quad (2)$$

As for the waveform example, the original $SNR = 11.35$. Then, Gaussian noise of certain energy is added to the filtered effective signal (the final SNR should be different from the SNR of the original signal), where the SNR is set as $SNR = 5$, and then new sample data are obtained. The waveform and spectrum of the new sample are shown in Figure 5.

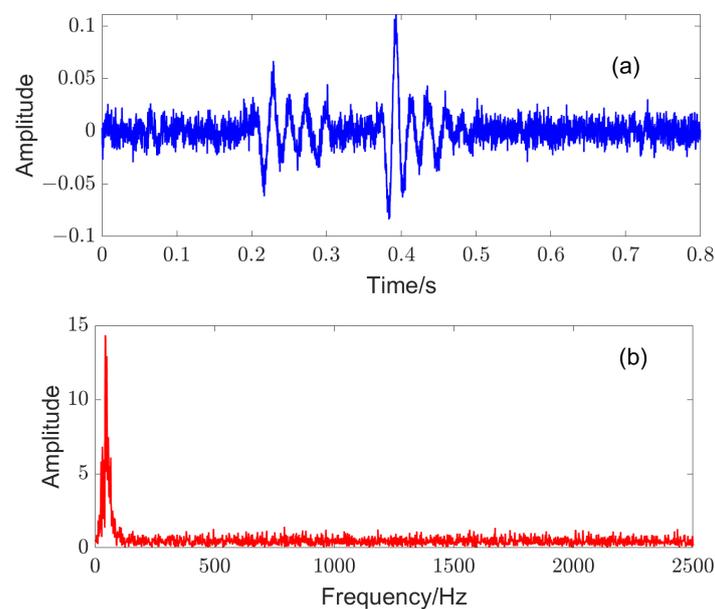


Figure 5. Signal waveform and spectrum of $SNR = 5$ after noise injection. (a) is the waveform, (b) is the spectrum.

The comparison of Figures 4 and 5 illustrates that adding Gaussian noise does not change the kind of waveform, and the method of increasing the number of samples by adding noise to the signal is correct.

To balance the data set, in the upsampling process, the bandpass-filtered samples of class 2 and class 3 are upsampled and injected with Gaussian noise of different energy four times according to the original SNR . The SNR of the events in class 2 and class 3 after noise injection is 2 dB, 3 dB, 4 dB, and 6 dB lower than the original SNR , respectively.

The number of training events at this time is as follows:

$$\begin{aligned} N_{n2} &= N_{n1} + 4 \times (N_2 + N_3) \\ &= 5988 + 4 \times (361 + 314) \\ &= 8688 \end{aligned} \quad (3)$$

The numbers of samples of various classes in the training set at this time are 4638, 2166, and 1884. After upsampling and adding noise, the data ratio of various types in the training set is close to 23:11:9, and the data balance has been greatly improved. The total number of samples in class 1 is still larger than the other two classes, therefore the balance still needs to be further improved.

2.3. Window Warping and Window Slicing

Window slicing is a commonly used data augmentation technique in the field of image recognition [37,38] which randomly selects image patches with fixed size but smaller than the original image as the training set. And it was introduced into time-series signal data augmentation intercepting time slices of specific length from time-series signals and classifying them at the slice level [39]. The size of the slice window is a key parameter in the window slicing.

Figure 6 demonstrates the diagram of the window slicing. Here, Figure 6b,c are the corresponding signal slices inside the red and blue boxes in Figure 6a, and the slice lengths are 75% and 87.5% of the original signal, respectively. Figure 6 clearly shows that each slice extracted from the same signal will be assigned the same label as the original signal.

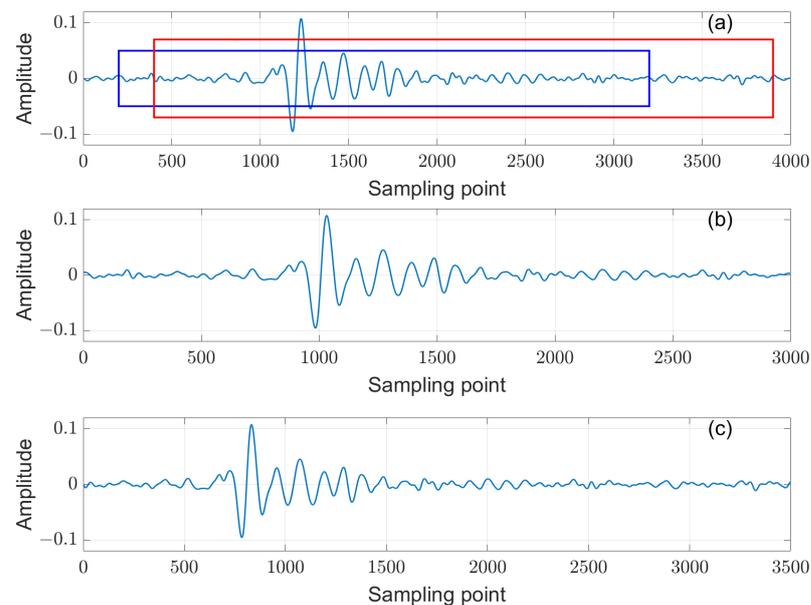


Figure 6. Example of the window slicing. (a) is the original signal, (b) is the slicing of blue box in (a), (c) is the slicing of red box in (a).

Window warping is a unique data augmentation technique for time series signals [31]. This method speeds up or slows down (deceleration or acceleration) the time-series signals in the slice window to achieve the enhancement of time-series data. A schematic diagram of the window warping method is shown in Figure 7. In the example, the slice window length is 400 (10% of the original signal length) and the warping rate σ is 2 and 0.5, respectively.

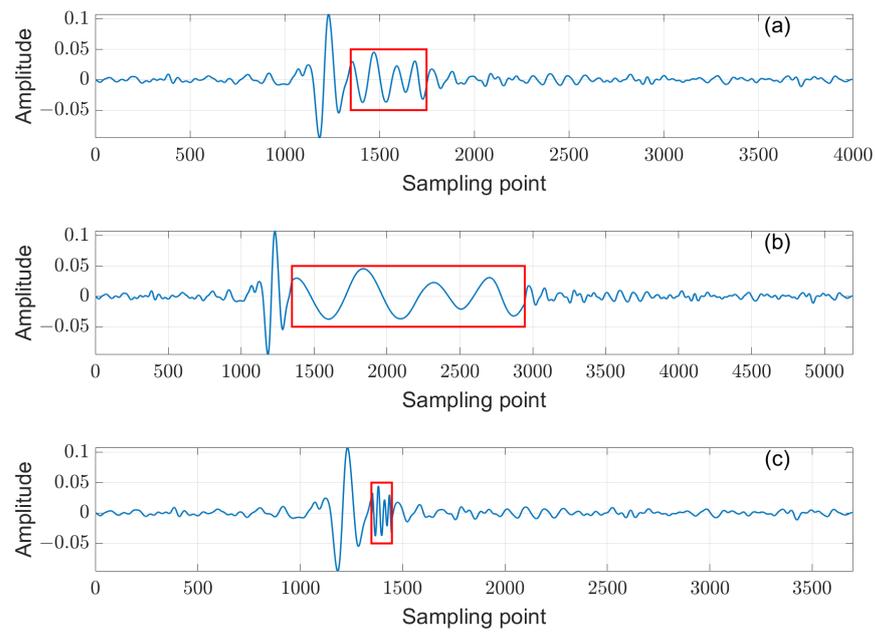


Figure 7. Example of the window warping, (a) is the original, (b) is deceleration, (c) is acceleration.

The length of the slice window and the warping rate are two parameters that affect the results of the window warping method. The relationship between the length of the timing signal after transformation and the length of the slice and the warping rate is determined.

$$N_{new} = \begin{cases} N_{ori} + (\sigma - 1) \times N & , \sigma < 1 \\ N_{ori} + (N - 1) \times \sigma + 1 & , \sigma > 1 \end{cases} \quad (4)$$

where N_{new} is the transformed signal length, N_{ori} is the original signal length, N is the slice length, and σ is the warping rate. We refer to the transformation process when the warping rate is greater than 1 as deceleration and to the transformation process when the warping rate is less than 1 as acceleration. Acceleration and deceleration also have different effects on the length of the signal: deceleration will increase the length of the signal, while acceleration will decrease it.

Window warping generates time series of different lengths, whereas machine learning requires the input data to be of the same size. To solve this problem, we can slice the transformed timing signal to keep it the same length.

By combining the window slicing and window warping, we can further increase the number of training samples. In this paper, we first use the window warping method to enhance the timing signal after upsampling, and then use the slice window method to extract fixed-length signal segments.

Here, the combination of three pairs of warping window length and warping degree is selected to perform warping operation on the original timing signal. The combination is shown in Table 1.

Table 1. The combination of warping window length and warping ratio.

Warping window length	200	300	500
Warping	5	4	3

The timing signals of different lengths after warping are sliced so that the length of all segments is 4000. For the transformation results corresponding to each parameter

combination, the signal with label 1 is sliced once, and the other two types of samples are sliced twice. The total number of samples is as follows:

$$N_{n3} == 3 * 4638 + 6 * 2166 + 6 * 1884 = 38214 \quad (5)$$

Currently, the numbers of samples of various types in the training set are 13,914, 12,996, and 11,304. The proportion of samples of various types is close to 1:1:1. Not only is the number of sample sets greatly increased but also the distribution balance is greatly improved, which can be used in the training set data of neural networks.

3. FCN-LSTM

Here, we try to add a convolution layer before the LSTM network, to build a hybrid network structure, and study its feasibility for the automatic classification for the microseismic signals. Figure 8 shows the structure diagram of the FCN-LSTM network containing one convolutional layer.

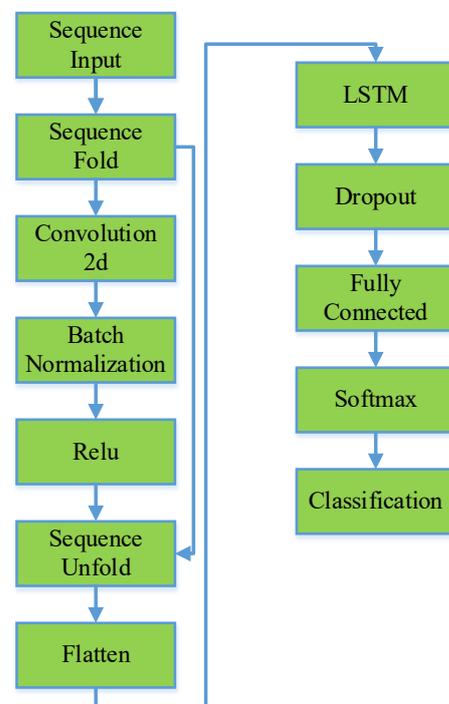


Figure 8. FCN-LSTM network structure with one convolution layer.

The network structure and some training parameters are set as follows:

1. The input layer: time series of length 4000.
2. The convolution layer: 32 convolution kernels of size 7×1 .
3. The LSTM layer: 32 hidden units.
4. Dropout: 0.1.
5. The initial learning rate: 0.0001.
6. Learning rate schedule: piecewise.
7. Shuffle: once.
8. Gradient threshold: 0.8.
9. Max epochs: 30.
10. Mini-batch: 128.
11. Validation frequency: every 50 iterations.

In the process of network training, we randomly select 20% and 10% from the training set obtained in the previous section as the validation set and the test set, respectively. To eliminate the interference of other factors in the results of machine learning as much

as possible, the same graphics card is used to train the related networks. The partial training results of this network are shown in Table 2. This training takes 9 min and 26 s for 6270 iterations. The final accuracy and final loss for the training set are 81.25% and 0.5329, respectively. The final accuracy and final loss for the validation set are 82.21% and 0.4892, respectively. It can be found that the final model trained by this network is convergent, and the accuracy of the training set and the validation set has even exceeded 80%, which can already be regarded as a relatively accurate model.

Table 2. Partial training results of network.

Number of Convolutional Layers	Max Epoch	Iteration Number	Time Taken	Final Accuracy of Training Set	Final Accuracy of Validation Set	Final Loss of Training Set	Final Loss of Validation Set
1	30	6270	0:09:26	81.25%	82.21%	0.5329	0.4892

Figure 9 is the variation curves of accuracy and loss of the training set and validation set during the training of the neural network. These curves indicate that the model obtained by training is convergent. It is feasible to use the FCN-LSTM network to realize the automatic classification of microseismic events.

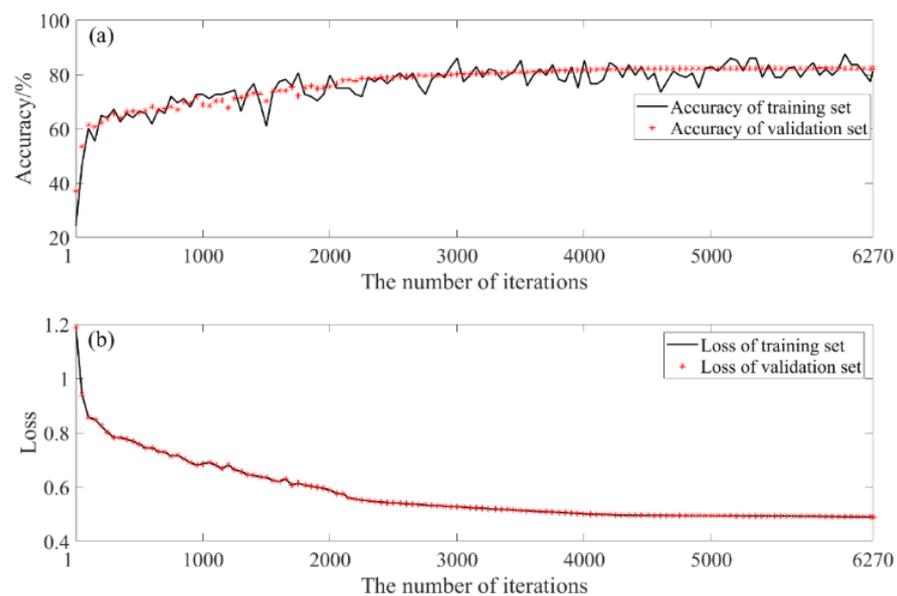


Figure 9. The training process of the FCN-LSTM network. (a) is the accuracy of the training set and validation set, (b) is loss of the training set and validation set.

Next, we study the influence of the number of convolution layers on the final model through experiments. The numbers of convolution layers are 3, 5, and 7, the size of each convolution kernel was 7×1 , and the number of convolution kernels in each layer of the three-layer convolution structure was 32, 16, and 32, respectively. The number of convolution kernels in each layer of the five-layer convolutional structure is 32, 16, 8, 16, and 32, respectively. The number of convolution kernels in each layer of the five-layer convolutional structure is 32, 24, 16, 8, 16, 24, and 32, respectively. The network structure is demonstrated in Figure 10.

Except for the number of convolutional layers and the number of epochs, the other training parameters of the three network structures were consistent with the network with one fully convolutional layer. The partial training results of three different network structures are shown in Table 3.

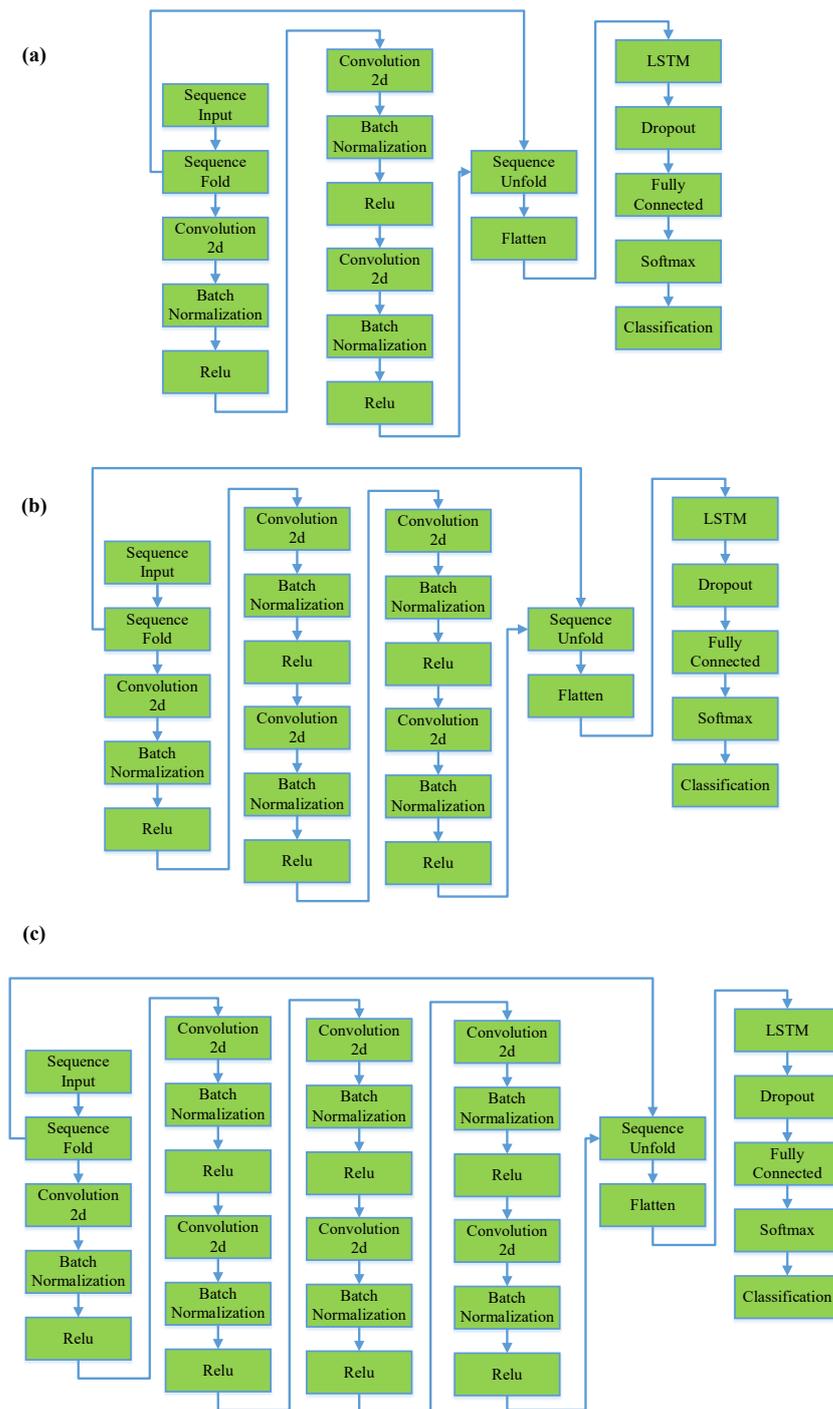


Figure 10. Schematic of the FCN-LSTM network structure with multiple convolutional layers. (a) is the network with 3 convolutional layers, (b) is the network with 5 convolutional layers, and (c) is the network with 7 convolutional layers.

Table 3. Partial training results of three networks.

Number of Convolutional Layers	Max Epoch	Iteration Number	Time Taken	Final Accuracy of Training Set	Final Accuracy of Validation Set	Final Loss of Training Set	Final Loss of Validation Set
3	25	5225	0:18:24	89.84%	90.33%	0.3606	0.3561
5	25	5225	0:23:35	89.84%	93.41%	0.3417	0.3141
7	25	5225	0:29:43	91.41%	90.36%	0.3464	0.3525

The training results of the three networks show some improvement over the training results of the previous network structure. By comprehensively comparing the training time, accuracy, and loss of the above three network structures, the FCN-LSTM network containing five convolutional layers is the best network among the three. Figure 11 is the variation curves of accuracy and loss for the training set and validation set during the training of the neural network. These curves indicate that the model obtained by training is convergent.

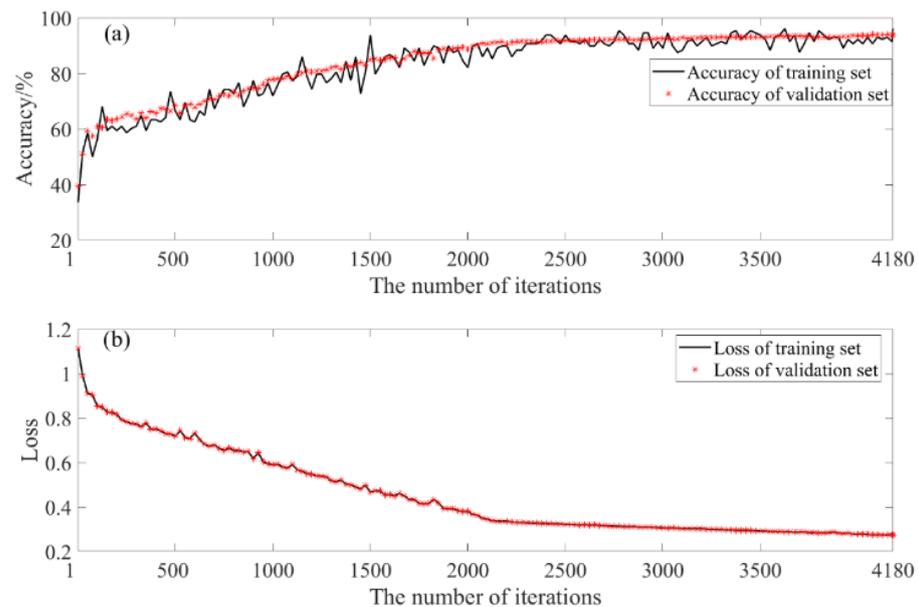


Figure 11. The training process of the FCN-LSTM network with 5 convolutional layers. (a) is the accuracy of the training set and validation set, (b) is loss of the training set and validation set.

In summary, the FCN-LSTM network with five convolutional layers can be regarded as the most ideal choice for realizing automatic classification of microseismic data.

The test set accuracy is calculated using the following formula:

$$Test-acc = \frac{\text{count}(CTest(i) = YTest(i))}{N} \quad 1 \leq i \leq N \quad (6)$$

where N is the number of samples in the test set, count represents the total number of classification labels identified by the neural network consistent with the test set labels, $CTest$ is the classification label identified by the neural network, and $YTest$ is the test set label.

For the whole test set, the number of correctly classified samples is 3465. According to Formula (6), the test set accuracy of the automatic classification method of microseismic events based on a five-convolutional-layer FCN-LSTM network reaches 91.18%, which can meet the requirements of actual production activities.

4. Conclusions

The intelligent classification of microseismic events is of great significance for coal mine disaster detection and early warning, and the classification method based on machine learning principles has been paid more and more attention by researchers because of its advantages in modeling. In this paper, the application of an FCN-LSTM network in the intelligent classification of microseismic events in coal mines is studied.

The available training set data are obtained by increasing the number of samples and balancing the sample distribution through bandpass filtering, upsampling, window slicing, and window warping. The FCN-LSTM hybrid network structure is built, and five convolutional layers are added in front of the LSTM structure. Then, 70%, 20%, and 10% of the enhanced data set are randomly selected as the training set, validation set, and test set,

respectively. Through training, the accuracy of the model validation set is 93.41%, and the accuracy of the validation set is 91.18%. The trained FCN-LSTM model is highly suitable. The first kind of microseismic event is related to underground water movement. The number and proportion of water inrush disasters markedly increase during its occurrence. Attention should be given to the number and proportion of these kinds of events during microseismic monitoring.

In this paper, we propose a real-time intelligent classification method for microseismic events based on data augmentation theory and machine learning theory. This method can obtain a highly suitable network model in the case of limited and unbalanced distribution of microseismic monitoring data, complete intelligent classification of microseismic events in real time, and have a high accuracy.

Author Contributions: Formal analysis, L.Z.; Resources, H.L.; Writing—original draft, G.S.; Writing—review & editing, G.S., D.L. and G.Q.; Supervision, L.L. and X.L.; Funding acquisition, D.L. and X.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was by the National Natural Science Foundation of China (52174218, 52204257) and the China Postdoctoral Science Foundation (2022M713371).

Data Availability Statement: Not applicable.

Conflicts of Interest: Authors Guojun Shang and Dexing Li were employed by both the China University of Mining and Technology and Shenzhen Urban Public Safety Technology Research Institute Co., Ltd., authors Li Li and Gan Qin were employed by the company Shenzhen Urban Public Safety Technology Research Institute Co., Ltd., author Liping Zhang was employed by the Dagang Oilfield Foreign Cooperation Projects Department of PetroChina. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Albright, J.; Pearson, C. Acoustic emissions as a tool for hydraulic fracture location: Experience at the Fenton hill hot dry rock site. *Soc. Pet. Eng. J.* **1982**, *22*, 523–530. [[CrossRef](#)]
2. Liang, B.; Shen, C.; Leng, C.; Guo, B.; Yang, Y.; Zheng, B. Development of microseismic monitoring for hydro-fracturing. *Prog. Geophys.* **2015**, *30*, 401–410.
3. Van Der Baan, M.; Eaton, D.; Dusseault, M. Microseismic monitoring developments in hydraulic fracture stimulation. In Proceedings of the ISRM International Conference for Effective and Sustainable Hydraulic Fracturing, Brisbane, Australia, 20–22 May 2013.
4. Phillips, W.; Fairbanks, T.; Rutledge, J.; Anderson, D. Induced microearthquake patterns and oil-producing fracture systems in the Austin chalk. *Tectonophysics* **1998**, *289*, 153–169. [[CrossRef](#)]
5. Rutledge, J.T.; Phillips, W.S.; Mayerhofer, M.J. Faulting induced by forced fluid injection and fluid flow forced by faulting: An interpretation of hydraulic fracture microseismicity, Carthage Cotton Valley gas field, Texas. *Bull. Seismol. Soc. Am.* **2004**, *94*, 1817–1830. [[CrossRef](#)]
6. Maxwell, S.C.; Rutledge, J.; Jones, R.; Fehler, M. Petroleum reservoir characterization using downhole microseismic monitoring. *Geophysics* **2010**, *75*, 129–137. [[CrossRef](#)]
7. Chen, Y.; Saad, O.M.; Savvaidis, A.; Chen, Y.; Fomel, S. 3D microseismic monitoring using machine learning. *J. Geophys. Res. Solid Earth* **2022**, *127*, e2021JB023842. [[CrossRef](#)]
8. Aster, R.C.; Shearer, P.M.; Berger, J. Quantitative measurements of shear wave polarizations at the Anza Seismic Network, southern California: Implications for shear wave splitting and earthquake prediction. *J. Geophys. Res.* **1990**, *95*, 12449–12473. [[CrossRef](#)]
9. Bao, X.; Eaton, D.W. Fault activation by hydraulic fracturing in western Canada. *Science* **2016**, *354*, 1406–1409. [[CrossRef](#)]
10. Chen, H.; Meng, X.; Niu, F.; Tang, Y. Microseismic monitoring of stimulating shale gas reservoir in SW China: 2. Spatial clustering controlled by the preexisting faults and fractures. *J. Geophys. Res.* **2018**, *123*, 1659–1672. [[CrossRef](#)]
11. Wang, Y.; Qiu, Q.; Lan, Z.; Chen, K.; Zhou, J.; Gao, P.; Zhang, W. Identifying microseismic events using a dual-channel CNN with wavelet packets decomposition coefficients. *Comput. Geosci.* **2022**, *166*, 105164. [[CrossRef](#)]
12. Jiang, F.; Miao, X.; Wang, C. Predicting research and practice of tectonic-controlled coal burst by microseism is monitoring. *J. China Coal Soc.* **2010**, *35*, 900–903.
13. Duan, D.; Tang, C.; Feng, X. Microseismic monitoring system establishment and its application to Xinzhangzi coal mine. *Adv. Mater. Res.* **2011**, *396–398*, 99–102. [[CrossRef](#)]

14. Zhang, C.; Jin, G.; Liu, C.; Li, S.; Xue, J.; Cheng, R.; Wang, X.; Zeng, X. Prediction of rockbursts in a typical island working face of a coal mine through microseismic monitoring technology. *Tunn. Undergr. Space Technol.* **2021**, *113*, 103972. [[CrossRef](#)]
15. Shang, G.; Liu, X.; Li, L.; Zhao, L.; Shen, J.; Huang, W. Cluster analysis of the domain of microseismic event attributes for floor water inrush warning in the working face. *Appl. Geophys.* **2023**, *19*, 409–423. [[CrossRef](#)]
16. Hardy, H.R. *Acoustic Emission/Microseismic Activity, Volume 1: Principles, Techniques and Geotechnical Applications*; Balkema Publishers: Amsterdam, The Netherlands, 2003.
17. Ge, M. Efficient mine microseismic monitoring. *Int. J. Coal Geol.* **2005**, *64*, 44–56. [[CrossRef](#)]
18. Li, D.; Wang, E.; Feng, X.; Wang, D.; Zhang, X.; Ju, Y. Weak current induced by coal deformation and fracture and its response to mine seismicity in a deep underground coal mine. *Eng. Geol.* **2023**, *315*, 107018. [[CrossRef](#)]
19. Li, D.; Wang, E.; Jin, D.; Wang, D.; Liang, W. Response characteristics of weak current stimulated from coal under an impact load and its generation mechanism. *Sustainability* **2023**, *15*, 2605. [[CrossRef](#)]
20. Hochreiter, S. Untersuchungen zu Dynamischen Neuronalen Netzen. Master's Thesis, Technical University of Munich, Munich, Germany, 1991.
21. Graves, A.; Mohamed, A.; Hinton, G. Speech recognition with deep recurrent neural networks. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013; pp. 6645–6649.
22. Gers, F.; Schmidhuber, J.; Cummins, F. Learning to forget: Continual prediction with LSTM. *Neural Comput.* **2000**, *12*, 2451–2471. [[CrossRef](#)]
23. Sutskever, I.; Vinyals, O.; Le, Q. Sequence to sequence learning with Neural Networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 3104–3112.
24. Karpathy, A.; Li, F. Deep visual-semantic alignments for generating image descriptions. In Proceedings of the 2015 Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 3128–3137.
25. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)]
26. Le Guennec, A.; Malinowski, S.; Tavenard, R. Data augmentation for time series classification using Convolutional Neural Networks. In Proceedings of the ECML/PKDD Workshop on Advanced Analytics and Learning on Temporal Data, Riva Del Garda, Italy, 19 September 2016.
27. Wen, Q.; Sun, L.; Song, X.; Gao, J.; Wang, X.; Xu, H. Time series data augmentation for deep learning: A survey. In Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, Montreal, QC, Canada, 19–27 August 2021; pp. 4653–4660.
28. Brain, D.; Webb, G. On the effect of data set size on bias and variance in classification learning. In Proceedings of the Fourth Australian Knowledge Acquisition Workshop (AKAW'99), Sydney, Australia, 5–6 December 1999; pp. 117–128.
29. Juba, B.; Le, H. Precision-Recall versus Accuracy and the Role of Large Data Sets. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 4039–4048.
30. Murphey, Y.; Guo, H.; Feldkamp, L. Neural learning from unbalanced data. *Appl. Intell.* **2004**, *21*, 117–128. [[CrossRef](#)]
31. Jaitly, N.; Hinton, G. Vocal Tract Length Perturbation (VTLP) improves speech recognition. In Proceedings of the 30th International Conference on Machine Learning, Atlanta, GA, USA, 16–21 June 2013.
32. Cui, X.; Goel, V.; Kingsbury, B. Data augmentation for deep neural network acoustic modeling. *IEEE/ACMT Trans. Audio Speech Lang. Process.* **2015**, *23*, 1469–1477.
33. Ko, T.; Peddinti, V.; Povey, D.; Khudanpur, S. Audio augmentation for speech recognition. In Proceedings of the Sixteenth Annual Conference of the International Speech Communication Association, Dresden, Germany, 6–10 September 2015; pp. 3586–3589.
34. Lotte, F. Signal processing approaches to minimize or suppress calibration time in oscillatory activity-based brain-computer interfaces. *Proc. IEEE* **2015**, *103*, 871–890. [[CrossRef](#)]
35. Krell, M.; Seeland, A.; Kim, S. Data augmentation for brain-computer interfaces. analysis on event-related potentials data. *arXiv* **2018**, arXiv:1801.02730.
36. Steven, E.O.; Han, D.S. Feature representation and data augmentation for human activity classification based on wearable IMU sensor data using a deep LSTM neural network. *Sensors* **2018**, *18*, 2892. [[CrossRef](#)]
37. Howard, A.G. Some improvements on deep Convolutional Neural Network based image classification. *arXiv* **2013**, arXiv:1312.5402.
38. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with Deep Convolutional Neural Networks. *Commun. ACM* **2017**, *60*, 84–90. [[CrossRef](#)]
39. Cui, Z.C.; Chen, W.L.; Chen, Y.X. Multi-Scale convolutional neural networks for time series classification. *arXiv* **2016**, arXiv:1603.06995.

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.