J. Meas. Sci. Instrum., 2024, 15(2): 264-275.
DOI: 10.62756/jmsi.1674-8042.2024027
http://jmsi.publish.founderss.cn
jmsi@nuc.edu.cn

Intelligent diagnosis method of rolling bearing based on BiGAN

ZHANG Hao, GU Lichen*, GUO Zichen

School of Mechanical and Electrical Engineering, Xi'an University of Architecture and Technology, Xi'an 710055, China

***Corresponding author:** GU Lichen (gulichen@126.com)

Received: September 10, 2023 **Revised:** October 25, 2023 **Accepted:** December 7, 2023

Abstract: Rolling bearing is a critical component in the rotating machinery, which directly affects the reliability of the equipment. The artificial intelligence-enabled bearing fault diagnosis model has achieved impressive successes over the years. However, rolling bearings' imbalanced data sets (normal samples are much larger than failure samples) degrade the diagnostic performance. To address this issue, a bidirectional generative adversarial network (BiGAN) based fault diagnosis method was proposed. First, the signal was denoised via the ensemble empirical mode decomposition (EEMD) to automatically distribute it to a suitable reference scale and avoid modal aliasing. Then, the BiGAN model with gradient penalty term was constructed to expand the fault samples, where the min-max normalization was included. Finally, based on the enhanced training set, the convolutional neural network was established with batch normalization and maximum pooling layers. Experimental results proved that the proposed method improved fault diagnosis accuracy and robustness.

Key words: rolling bearing; fault diagnosis; bidirectional generative adversarial network (BiGAN); convolutional neural network (CNN); data imbalance

0 Introduction

Rolling bearing is essential for ensuring the safe operation of the large rotating machinery[1]. Notably, 50% — 60% of rotating machinery faults are caused by bearing faults. Therefore, the state detection and fault diagnosis of bearings are the main issues of current research[2]. There are broadly two general categories for fault diagnosis methods, including fault diagnosis based on mechanism knowledge and expert experience and data-driven fault diagnosis. However, if the fault diagnosis method is only based on knowledge and experience, it has limitations regarding non-stationary operating conditions, weak fault characteristics, and mixed fault modes[3].

With the vigorous development of the industrial internet and big data science, it is possible to diagnose faults under complex working conditions of rolling bearings. Therefore, data-driven bearing fault diagnosis methods have emerged. In addition, deep learning has more advantages than traditional shallow learning methods in terms of adaptive feature learning ability and multi-layer nonlinear mapping ability[4]. Jin et al.[5] proposed a rolling bearing fault diagnosis method based on a two-dimensional image convolutional neural network (CNN) structure. Zhao et al.[6] proposed a rolling bearing diagnosis method with one-dimensional CNN. The obtained spectral information is one channel to the diagnostic model in the signal preprocessing stage. Han et al.[7] combined a generative adversarial network with CNN, which effectively improved the robustness of feature identification and the model's generalization ability. Hao et al.[8] proposed an end-to-end pipeline for bearing fault diagnosis based on 1D-CLSTM. To sum up, the CNN-based fault diagnosis methods have achieved great success.

However, the data imbalance problem is often ignored in practical applications, such as aerospace and other high-end equipment fields. In addition, most equipment works under normal operating conditions[9]. The data imbalance problem manifests in insufficient fault types and sample numbers, resulting in fewer training samples, which eventually leads to the overfitting of the CNN model and low diagnostic accuracy[10].

Therefore, resolving data set imbalance and balancing various types of faults in the data set is the research focus of deep learning applied to fault diagnosis[11]. The general approaches to the data imbalance problem fall into two main categories. The first is based on cost-sensitive learning[12], which improves model accuracy with a small amount of sample data, and the main problem is that the loss of misclassification cannot be determined. It is challenging to

assess model validity. The second category focuses on the data preprocessing, such as oversampling or undersampling[13]. However, the oversampling technique increases the number of samples without incremental data features input into the model, and the model overfitting problem remains unmitigated. Data undersampling leads to data information loss and reduces diagnostic accuracy. Therefore, how to mine potential fault features through unbalanced one-dimensional vibration data is urgently needed to be studied. Generative adversarial network (GAN)[14] generates new data samples by simultaneously optimizing the generator and discriminator, and it is widely used in computer vision, natural language processing, and human-computer interaction. Lee et al.[15] used GAN to generate faulty samples to solve the imbalanced data problem. Wang et al.[16] constructed a new GAN discriminator in gearbox fault diagnosis using stacked denoising autoencoders as GAN discriminators and introduced category labels to achieve small-sample gearbox fault diagnosis.

A BiGAN-based rolling bearing fault diagnosis method was proposed, a bi-directional generative adversarial network containing gradient penalty terms was constructed to expand the fault samples, fault diagnosis was achieved by a deep convolutional model, and the effectiveness of the method was verified.

1 Methodology

1.1 Ensemble empirical mode decomposition

Empirical mode decomposition (EMD) is a nonlinear non-stationary signal decomposition method that decomposes a time series into multiple empirical modes called intrinsic mode functions (IMFs)[17], each IMF representing a narrowband amplitude-frequency signal corresponding to a specific physical process. Ensemble empirical mode decomposition (EEMD) is a significant improvement of the EMD method[18], which first adds finite-amplitude white noise to the original data to generate an integrated dataset and applies EMD to each time series in the integrated dataset by averaging over several obtained IMFs, which is implemented as follows.

1) The original signal $x(t)$ is added with random white noise signal $n_j(t)$, that is

$$x_j(t)=x(t)+n_j(t), \tag{1}$$

where $x_j(t)\left(j=1,2,\cdots,M\right)$ represents the added noise signal; M represents the number of experiments.

2) The signal $x_j(t)$ is decomposed into a series of eigenmode functions by EMD, namely $c_{i,j}$.

$$x_j(t)=\sum_{i=1}^{N_j}c_{i,j}+r_{N_j}, \tag{2}$$

where $c_{i,j}$ represents the ith IMF in the jth experiment; r_{N_j} represents the residual of the jth experiment; and N_j represents the number of IMF in the jth experiment. If $j<M$, repeat the two steps above, adding a different white noise signal to the original data signal each time.

3) Obtain $I=\min\left(N_1,N_2,\cdots,N_M\right)$, and calculate the average value of the corresponding IMF of the integrated dataset as the final decomposition result, i.e.,

$$c_i=\left(\sum_{j=1}^{M}c_{i,j}\right)\Big/M,\ \ i=1,2,\cdots,I, \tag{3}$$

where c_i denotes the integrated average of IMF.

The interrelationships between the decomposed each eigenmode component (IMF) and the original signal is quantified as

$$\rho_{s,\tilde{c}_j}=\max\left[R_{s,\tilde{c}_j}(\tau)\right]\Big/\max\left[R_s(\tau)\right], \tag{4}$$

where $R_{s,\tilde{c}_j}(\tau)$ is the intercorrelation between IMF and the original signal; $R_s(\tau)$ is the autocorrelation of the original signal. The higher the index, the more significant the correlation with the original signal.

Calculate the kurtosis index for each IMF, that is

$$K=\frac{E\left(x-\mu\right)^4}{\sigma^4}, \tag{5}$$

where μ and σ are the mean and standard deviation of the signal x, respectively; E represents the expected value.

The larger the IMF kurtosis value and the correlation coefficient, the higher the similarity with the original signal, which is used as the basis for signal reconstruction to achieve signal noise reduction. The noise-reduced signal is obtained through EEMD processing. The signal is subjected to fast fourier transform (FFT) to obtain the frequency domain signal. Then the signal is randomly divided into training set or test set samples.

1.2 Bidirectional generative adversarial network (BiGAN)

Generative adversarial network (GAN) usually consists of two parts: generator G and discriminator D. Both of them are neural networks with convolutional and fully connected layers. GAN can generate data samples through random latent variables, but there is no inverse mapping from a given training data x to the corresponding latent variable o. As shown in Fig. 1, an encoder component E on top of GAN is added to realize the mapping ($x\rightarrow o$) of the training data to the hidden

variables by bidirectional generative adversarial network (BiGAN). The principle is similar to that of a autoencoder, where the generator is equivalent to the decoder of the autoencoder ($o \rightarrow x$). Not only does the discriminator receive x as an input, but the hidden layer variables are also fed into the discriminator. For real samples, x comes from the given training set and o is generated by encoder E. For fake samples, o is randomly generated by a given prior distribution, and x is generated by generator G.

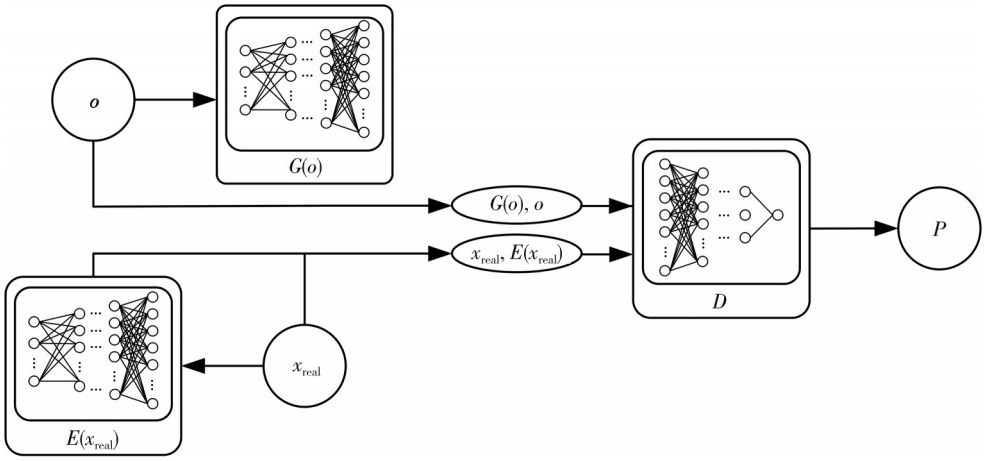

Fig. 1 Simplified architecture of BiGAN

The encoder E is also implemented by a deep neural network, and the training process is similar to that of the generator. The expression of the likelihood function of BiGAN is

$$L = \sum_{x^t} \log D\left(x^t, E\left(x^t\right)\right) + \sum_{o^t \sim p(o)} \log\left(1 - D\left(G\left(o^t\right), o^t\right)\right), \tag{6}$$

where x^t represents the real sample in training set x_{real}; o^t is a random distribution; $p(o)$ is the prior probability distribution the discriminator discriminates both the input sample data and its hidden variables.

Wasserstein distance is defined as the minimum cost to transform from one distribution to another, and the training process can be stabilized in generative adversarial networks by minimizing the Wasserstein distance, i. e., the distance between the generated samples and the true sample distribution, and the Wasserstein loss function is defined as

$$L_{\text{WGAN}} = \sum_{x^t} D\left(x^t\right) - \sum_{o^t \in p(o)} D\left(G\left(o^t\right)\right). \tag{7}$$

The discriminator D in the GAN model acts as a 0/1 classifier to estimate the posterior probability that the input is a real sample and outputs the discriminant through a nonlinear sigmoid function. The discriminator in the Wasserstein-GAN is a regressor to estimate the "true" score of the input, and outputs it through a linear function. Eq. (7) represents the score difference between the real and generated samples, and the model is trained to maximize the score difference while the generator is trained to minimize the score difference by introducing the Wasserstein distance[19] into BiGAN, which is redefined as

$$L_{\text{WBiGAN}} = \sum_{x^t} D\left(x^t, E\left(x^t\right)\right) - \sum_{o^t \in p(o)} D\left(G\left(o^t\right), o^t\right). \tag{8}$$

In order to improve the stability of BiGAN and the reliability of generating fault samples, a bidirectional generative adversarial network with gradient penalty was constructed. A small number of collected fault samples were used to train BiGAN alternately, and a single-sample outlier normalization method was designed to stabilize the training process. The similarity index was constructed to screen the generated fault samples and finally achieve the expansion of the fault samples. The Wasserstein-1 distance and gradient penalty method is used to stabilize the training process, and Wasserstein-1 distance is defined as

$$W(A_1, A_2) = \min_{\gamma \sim \Pi(A_1, A_2)} E_{(x,y)\sim\gamma}[\|x - y\|], \tag{9}$$

where A_1 is the real data distribution; A_2 is the generated data distribution; $W(A_1, A_2)$ is the set of joint distributions of A_1 and A_2; γ is one of the joint distributions; (x, y) is a pair of samples in γ; and $E_{(x,y)\sim\gamma}\left[\left\| x - y \right\|\right]$ is the expected value of the sample distance.

Based on the Wasserstein-1 distance[20] and the Kantorovich-Rubinstein dyadic principle, the model training process is represented as

$$M = \min_{G} \max_{D \in \Omega} E_{(x_r, z)\sim p_{x_r z}}[D(x_r, z)] - E_{(x_g, o)\sim p_{x_g o}}[D(x_g, o)], \tag{10}$$

where Ω is the set of 1-Lipschitz functions; $E_{(x_r,z)\sim p_{x_r z}}$ is the expected value from the real data distribution and $E_{(x_g,o)\sim p_{x_g o}}$ is the expected value from the noisy data distribution.

The loss function of the bidirectional generative adversarial network model can be expressed as

$$L = M + \lambda E_{(\hat{x},\hat{z})\sim p_{\hat{x}\hat{z}}}[(\|\nabla D(\hat{x},\hat{z})\|_2 - 1)^2], \quad (11)$$

where $(\hat{x},\hat{z})$ is sampled on the line of paired points of (x_r,z) and (x_g,o); $p_{\hat{x}\hat{z}}$ is its corresponding data distribution; and $\lambda E_{(\hat{x},\hat{z})\sim p_{\hat{x}\hat{z}}}[(\|\nabla D(\hat{x},\hat{z})\|_2 - 1)^2]$ is the gradient penalty terms. Compared with traditional generative adversarial network model, the optimized model has higher training stability.

The similarity index C_{in}[21,22] is constructed to screen the generated fault samples, where the similarity index is calculated by

$$C_{\text{in}} = \text{diag}\{FC, FSD\}, \quad (12)$$

where FC is the center of gravity frequency; FV is the frequency variance; and FSD is the frequency standard deviation. The equations are calculated by

$$FC = \frac{\sum_{t=0}^{N} f_t A(f_t)}{\sum_{t=0}^{N} A(f_t)}, \quad FV = \frac{\sum_{t=0}^{N} (f_t - FC)^2 A(f_t)}{\sum_{t=0}^{N} A(f_t)},$$
$$FSD = \sqrt{FV}, \quad (13)$$

where $A(f_t)$ is the amplitude and f_t is the frequency.

The Euclidean distance between the real sample and the generated sample is calculated by Eq. (14). If the difference between the calculated result and the difference is slight, it is considered that the generated fault sample is highly correlated with the real fault sample. The generated fault sample can be used as the supplementary data of the real fault sample.

$$C_{\text{index}}^{\text{test}} = \left\| C_{\text{in}}^{\text{real}} - C_{\text{in}}^{\text{generate}} \right\|_2,$$
$$C_{\text{indes}} = \max\left\{ \left\| C_{\text{in}}^{\text{real}_i} - C_{\text{in}}^{\text{real}_j} \right\|_2, i = 1,2,\cdots,n; j = 1,2,\cdots,n \right\}. \quad (14)$$

1.3 Convolutional neural network

The structural model of convolutional neural network (CNN) is shown in Fig.2.

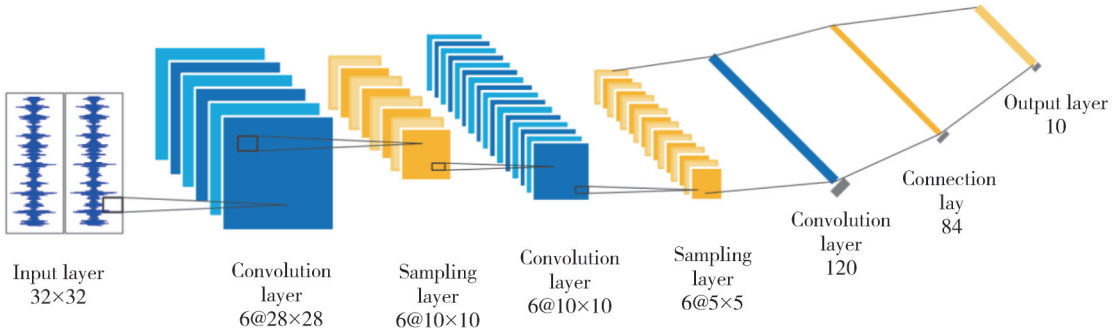

Fig. 2 Convolutional neural network model

Its forward propagation convolutional layer expression is

$$x_k^l = b_k^l + \sum_{i=1}^{N_{l-1}} \text{conv1}D(w_{ik}^{l-1}, s_i^{l-1}), \quad (15)$$

where b_k^l represents the deviation of the kth neuron; s_i^{l-1} represents the output of the ith neuron in the $l-1$ layer; and w_{ik}^{l-1} represents the weight value between the ith neuron in the $l-1$ layer and the kth neuron in the l layer.

The forward propagation down-sampling layer expression is

$$y_k^l = f(x_k^l), \; s_k^l = y_k^l \downarrow ss, \quad (16)$$

where $\downarrow$ represents the down-sampling process.

The fault diagnosis model described in this paper included at least four convolutional layers to achieve neural network weight sharing, and a batch normalization layer was added after each convolutional layer to improve the training speed and model generalization ability. The activation function was set to ReLU, and the loss function was set to the cross-entropy loss function.

Multiple convolution kernels were used to convolve the input samples. After adding the bias term, the signal features were obtained after the activation function. The mathematical expression of convolution is

$$X_j^l = f\left(\sum_{i \in M_j} X_i^{l-1} \boldsymbol{w}_{ij}^l + b_j^i \right), \quad (17)$$

where X_j^l is the jth element of the l layer; M_j is the jth convolution region of the $l-1$ layer feature map; $\boldsymbol{w}_{ij}^l$ is the corresponding weight matrix; b_j^i is the bias term; $f(\cdot)$ is the activation function. The convolutional neural network model implements the classification task by training the values of the weight matrix as well as the values of the bias terms.

The maximum pooling method maximizes the feature

map output from the convolutional layer in each non-overlapping size region $n \times n$.

The feature map is expanded into a one-dimensional feature vector, which is weighted and summed by the activation function.

$$y^k = f(w^k x^{k-1} + b^k), \tag{18}$$

where k is the ordinal number of the network layer; y^k is the output of the fully connected layer; x^{k-1} is a one-dimensional feature vector; w^k is the weight coefficient; b^k is the bias term.

The back-propagation algorithm was used to train the fault diagnosis model, and the gradient of the loss function for each weight was calculated by using the chain derivative. The weights were updated according to the gradient descent algorithm. The cost function used to solve the convolutional neural network is the cross-entropy function, which is given by

$$C = -\frac{1}{n}\sum_x [\, y \ln a + (1-y)\ln(1-a)\,], \tag{19}$$

where c represents the cost; x represents the sample; n represents the total number of samples; a represents the model output value; and y represents the actual value of the sample.

2 Intelligent diagnosis method based on BiGAN for rolling bearing

The diagnosis method flow in this paper is shown in Fig. 3. First, the signal was denoised via EEMD-based threshold filtering. On this basis, the signal was sequentially sampled with a length of 1 024 without overlapping the signal and subjected to FFT to obtain the frequency domain signal, then normalized by deviation. The signal was randomly divided into training set or test set samples. Second, the BiGAN model with a gradient penalty term was constructed. A small number of fault samples were alternated to train the model, and the single sample outlier normalization method was used to stabilize the training process.

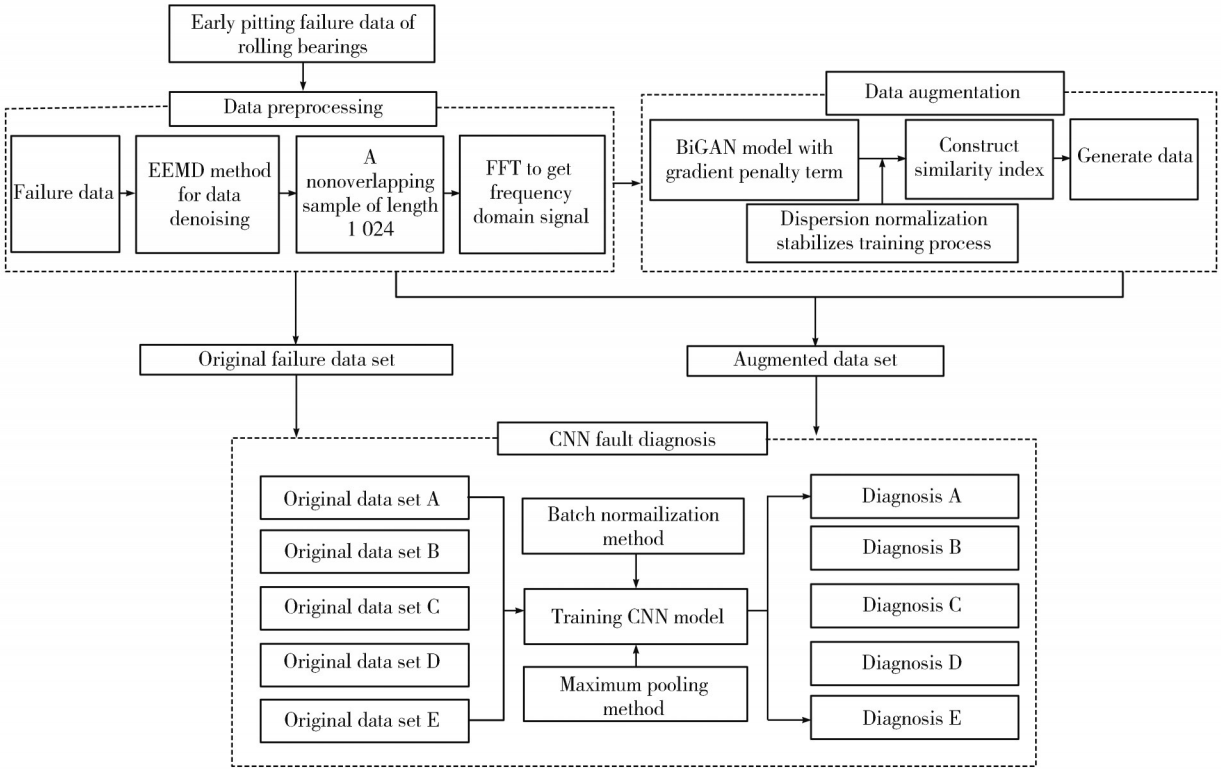

Fig. 3 Fault diagnosis flow of proposed method

The approach effectively improved the convergence difficulties of generative adversarial network models. When the training model reached Nash equilibrium, the unusable generative samples could be filtered out based on the similarity index and achieve fault sample expansion. Then, the original fault and augmented samples were mixed into the training data.

The convolutional neural network model was constructed with batch normalization and max pooling. The diagnostic accuracy under different data sets was compared. Experimental results proved that the proposed method improved fault diagnosis accuracy and robustness. The process of the rolling bearing fault diagnosis method based on BiGAN data enhancement is shown in Fig.4.

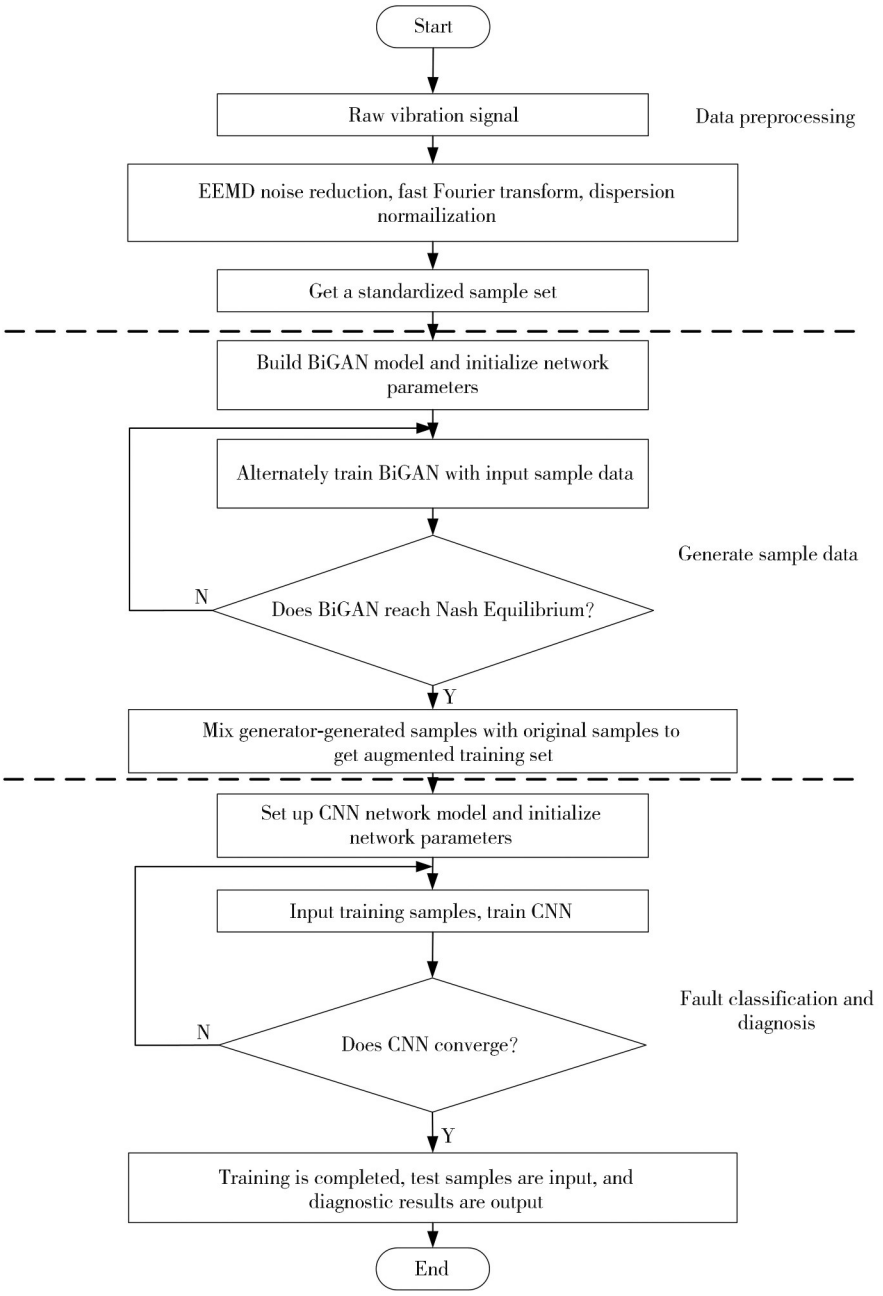

Fig. 4 Rolling bearing diagnosis method based on BiGAN

3 Data description

The test data were obtained from the bearing center database of case western reserve university (CWRU). The test setup consisted of a torque sensor, power meter, three-phase asynchronous motor, fan end bearing, and drive end bearing. The sampling frequency was 12 kHz. The fault types included inner ring fault, outer ring fault, and rolling element fault of the bearings. In addition, the fault damage diameters were 0.18 mm and 0.36 mm, respectively.

Six types of fault data and one type of normal data are used in this test, as shown in Table 1, which is divided into three data sets A, B, and C.

Table 1 Failure type of test data

Type	Fault type marker	Fault diameter/mm	Dataset A	Dataset B	Dataset C
Inner ring	0	0.18	200	2 000	858
	1	0.36			
Outer ring	2	0.18	400	2 000	858
	3	0.36			
Rolling element	4	0.18	600	2 000	858
	5	0.36			
Normal	6	0.00	2 000	2 000	858

Dataset A was mainly used to train BiGAN to generate fault sample data, containing a small amount of initial sample data of various faults. As shown in Fig.5, in order to make the statistical feature distribution of a small number

of failure samples obey the overall failure feature distribution, every 1 024 time-domain data points are used as a sample in this experiment.

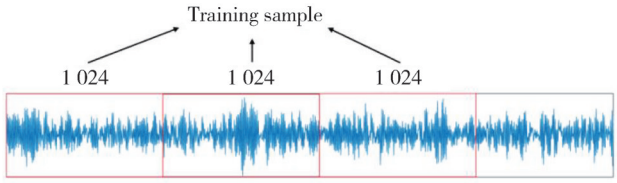

Fig. 5 Non-overlapping sampling

Dataset B was used to train CNN, which contained many faulty samples. Dataset C was used to test the training effect of the model. The two types of datasets were divided in the ratio of 7∶3, and their sample features were selected in the same way as dataset A, both using the sampling method.

The experiment was based on the open-source deep learning framework Pytorch to build the diagnosis model. It was divided into two parts, i.e., BiGAN to generate the fault sample balance dataset and CNN to perform fault diagnosis in two phases. The structural parameters of BiGAN model are shown in Table 2. Generator in BiGAN had three fully connected layers, and the hidden layer consisted of 256 neurons. Discriminator had four fully connected layers and two hidden layers consisting of 256 and 512 neurons. Encoder had three fully connected layers, and the hidden layer consisted of 256 neurons. The training batch size was set to 50, the initial learning rate is set to 0.001, and the number of iterations was 20. The activation function of the network in the generator was set to the sigmoid function.

Table 2 Structural parameters of BiGAN model

Parameter	Description	Dimensions	Weight parameter
G_in	Generator input dimension	200	
G_h	Generators hidden layer dimension	256	200×256
G_out	Generator output layer dimension	1 024	256×1 024
D_in	Discriminator input dimension	1 024	
D_h_1	Discriminator hidden layer dimension	256	1 024×256
D_h_2	Discriminator hidden layer dimension	512	256×512
D_out	Discriminator output layer dimension	1	512×1
E_in	Encoder input dimension	1 024	
E_h	Encoders hidden layer dimension	256	1 024×256
E_out	Encoder output layer dimension	200	256×200

The structural parameters of the network model in CNN are shown in Table 3. The model had four convolutional layers. The first convolutional layer consisted of 16 convolutional kernels, the second convolutional layer had 32 convolutional kernels, the third convolutional layer had 64 convolutional kernels, and the fourth convolutional layer had 128 convolutional kernels. The batch normalization layer was added after each convolutional layer, the activation function was set as ReLU and the loss function was the cross-entropy function.

Table 3 Structural parameters of convolutional neural network model

Network layer	Kernel size	Step	Number of kernels	Output size
Convolution layer	15×1	1×1	16	1 010×16
Convolution layer	3×1	1×1	32	1 008×32
Max pooling layer	2×1	2×1	32	504×32
Convolution layer	3×1	1×1	64	502×64
Convolution layer	3×1	1×1	128	500×128
Adaptive pooling			128	4×128
Fully connected layer	256		256	256
Fully connected layer	64		64	64
Softmax	7		1	7

4 Results and discussion

4.1 Model configuration and data preprocessing

During the experiment, four optimizers were tested respectively based on dataset B. The trend of recognition rate change during the training process of 20 epochs is analyzed as shown in Fig. 6, which reached stability when the number of iterations was 20 and could better reflect the classification ability of the network. The convergence speed of its recognition rate was Adam, RMSprop, Adagrad, and SGD in order from high to

low, so the Adam was chosen as the optimizer.

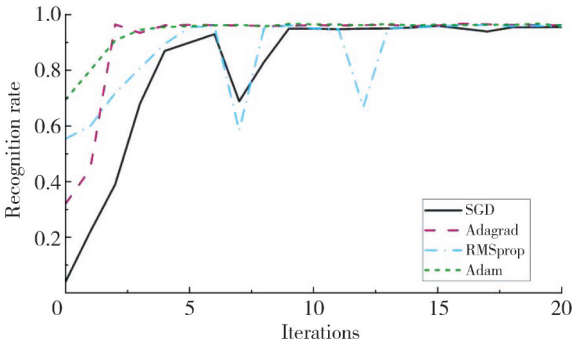

Fig. 6 Fault recognition rate curves for different optimizers

When constructing a neural network to fit the data, overfitting was easy to occur, resulting in poor generalization of the model, so regularization was needed to reduce the complexity of the model and reduce the testing error. The regularization methods such as L2 parametric, Dropout, and Batch Normalization (BN) were compared in this paper. As shown in Fig. 7, the convergence speed of the BN curve is significantly higher than other methods, and the recognition rate reached more than 90% after the 1st epoch, which indicated that BN could significantly improve the learning efficiency.

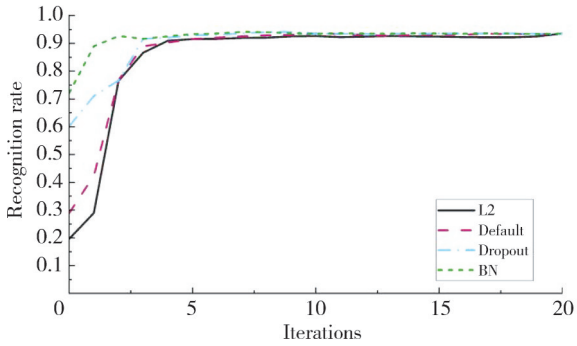

Fig. 7 Fault recognition rate curves for different regularization methods

EEMD decomposition for rolling bearing normal state (multiple samples) and fault state (few samples) was used to realize signal noise reduction. The original signal of rolling bearing inner ring pitting fault and the result after EEMD decomposition in CWRU University Bearing Data Center Library are shown as Fig.8.

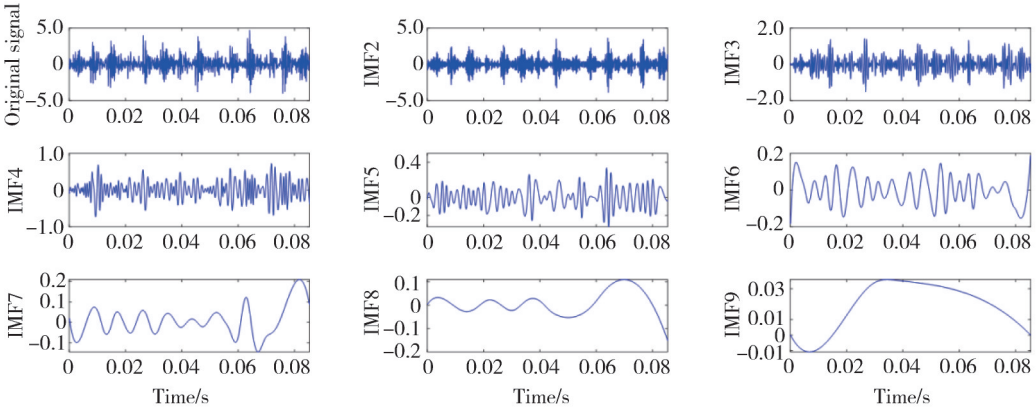

Fig. 8 Original signa and EEMD decomposition results

4.2 Comparison of generated and real fault samples

To verify the effectiveness of the BiGAN model in solving the data imbalance problem, each type of fault sample in the above data set A was input into BiGAN separately for training. When BiGAN reaches Nash equilibrium, the generated samples of 2 types of faults (inner ring pitting and ball pitting) have similarity indexes, as shown in Table 4. It was obvious that the generated fault samples had a high correlation with the original samples. The generated fault samples could be considered as the supplementary data of the real samples.

Similarly, from the comparison of the amplitude and frequency of the generated and real fault samples in Fig.9, the generated fault samples are similar to the real fault samples in the low and high-frequency bands, and the energy distribution of the frequencies in each band is also identical to the real fault samples.

Table 4 Generated fault sample accuracy metrics

Fault types	Sample type	FC	FSD
Inner ring pitting	Real sample 1	2 743.1	1 106.1
	Real sample 2	2 849.4	1 106.3
	Generate samples	2 722.3	1 023.2
Ball pitting	Real sample 1	2 853.6	1 108.4
	Real sample 2	2 760.3	1 108.9
	Generated samples	2 749.1	1 030.3

However, the spectral energy of the generated signal was not as concentrated as the real signal, and some of the sideband peaks were slightly different, which might be because the generated signal had more noise. The generated vibration signal could be regarded as the real vibration signal with more noise added.

The generated data samples were mixed with the original samples to obtain the mixed sample data, and

the CNN was trained by using the mixed sample data and the original sample data, respectively.

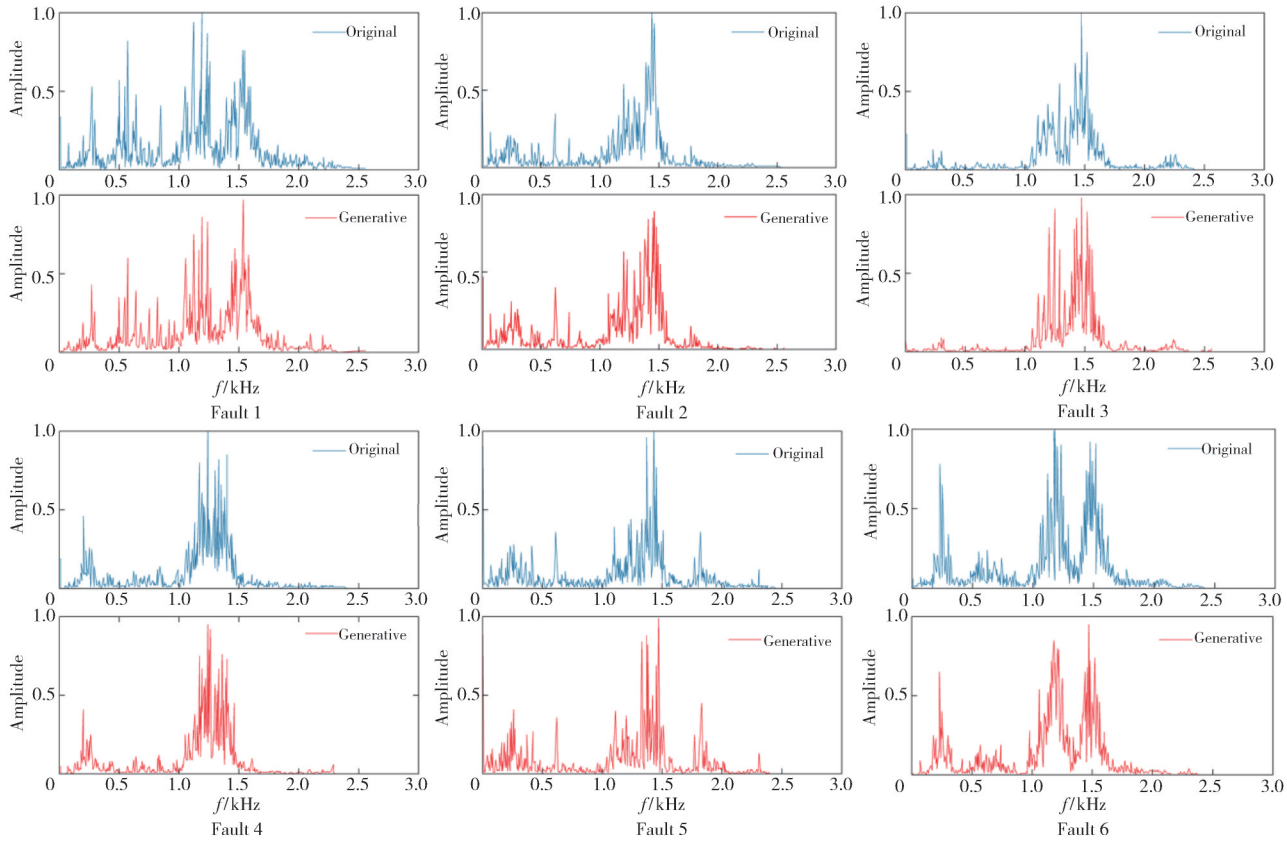

Fig. 9 Comparison of generated samples and real failure samples

And the effectiveness of the proposed method was verified by comparing the diagnostic accuracy of both under the same test set. The sample data set information is shown in Table 5. The normal state samples were not added to the generated samples, and the number of constant was 237.

Table 5 Diagnostic accuracy of test set with different training samples

Sample	Actual fault samples	Generated fault samples	Accuracy/%
A	204	0	60.42
B	224	0	81.25
C	257	0	89.58
D	329	0	97.92
E	204	72	95.83

It can be seen from Table 5 and Fig. 10 that the diagnostic accuracy of the model trained in samples *C* and *D* by the research method proposed in this paper was significantly higher than that of the training models in datasets *A* and *B*.

The data also showed that in the diagnosis of real fault samples, after the number of training samples was increased, the accuracy of the fault diagnosis model was significantly improved. However, most of the equipment was in normal operating conditions in practical applications.

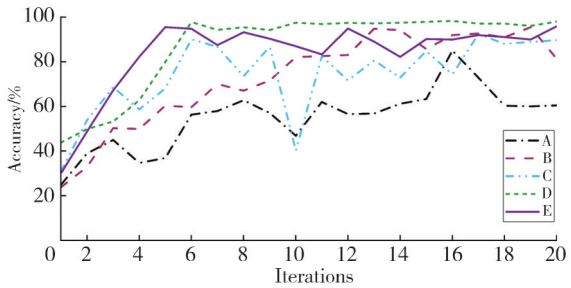

Fig. 10 Comparison of accuracy of BiGAN-based fault diagnosis methods

Therefore, it can be seen that due to the data imbalance problem caused by the lack of fault types and the number of samples, the CNN model will be over-fitted, and the diagnosis accuracy will be below. Ultimately, it is difficult to achieve the ideal fault diagnosis effect. After the BiGAN data enhancement method, the fault signal was obtained, the similarity index was compared, and the preliminary judgment was similar to the real fault. It could be used as a supplementary dataset for real faults. After experiments, it was found that under the enhanced data set *E*, the diagnostic accuracy of the trained model was 95.83%. In addition, the number of real fault samples was 204, and the number of generated fault samples was 72. The number of real fault samples in dataset *E* was consistent

with dataset A.

However, the diagnostic accuracy was increased by 35.41%. This showd that the data generated by BiGAN could complement the real fault samples. Comparing the number of real fault samples in dataset D, dataset E, and the number of enhanced mixed samples, the results showed that they were all significantly less than dataset D. However, the difference in the diagnostic accuracy of the models trained by the two was only 2.09%. This also further illustrated the reliability of generating fault samples as supplementary data.

4.3 Comparison results with other typical methods

The diagnostic accuracy of BiGAN was compared with other diagnostic models at different signal-to-noise ratios and different sizes of training sample. Three other diagnostic models were BPNN, ELM, and SVM. The number of hidden layer nodes for BPNN was set to 20. The radial basis kernel was selected as the kernel function of the SVM, and the cross-validation method was used to optimize the kernel function parameters and the penalty factor. The number of neurons in the implicit layer of the ELM was 100. Ten times fault diagnosis experiments were carried out. The average values of the fault diagnosis test results are taken as shown in Figs.11 and 12. It could be seen from Fig.11, as the signal-to-noise ratio decreased, the diagnostic performance of different diagnostic models showed a downward trend. At the same time, the diagnostic accuracy of BiGAN under different signal-to-noise ratios was above 90%. This is better than the other three diagnostic models. The results indicated that the noise robustness against noise of BiGAN was optimal under strong noise environment.

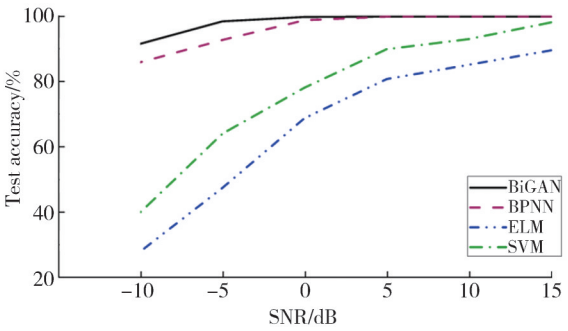

Fig. 11 Fault diagnosis results of four diagnosis models under different signal-to-noise ratios

In experiments with fewer diagnostic samples, the training samples for each state pattern were gradually reduced to simulate a small number of rolling bearing fault diagnoses. As shown in Fig.12, with the increase in the number of training samples, the diagnostic accuracy of the deep learning network model is gradually improved while the diagnostic accuracy of the shallow learning network models did not change that much. Compared with shallow learning network models, deep learning network models could dig out essential information from big data and exactly classify state modes of rolling bearing. Moreover, BiGAN achieved the highest diagnosis accuracy in the case of small training samples.

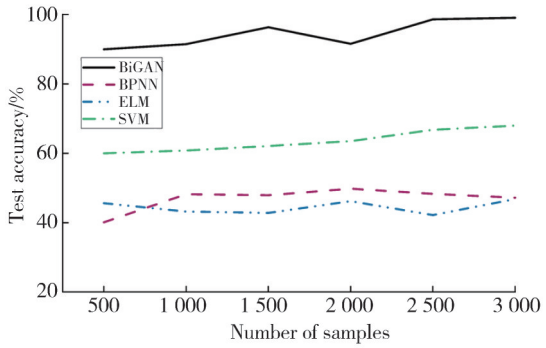

Fig. 12 Fault diagnosis results of four diagnosis models under different sample sizes

Based on this, it can be seen that the fault diagnosis method for rolling bearings based on BiGAN data enhancement can effectively use the bidirectional adversarial generation network to generate fault samples that are similar to the real samples. The comparison of experimental data verified the authenticity and reliability of the generated fault samples. It further solved the problem of low model diagnostic accuracy due to data imbalance.

5 Conclusions

A rolling bearing fault diagnosis method was proposed based on BiGAN data enhancement. It solves the problem of poor classification accuracy of diagnostic models due to unbalanced monitoring data for most high-reliability equipment in practical industrial scenarios. The proposed method was capable of learning the fault data characteristic distribution, thus enhancing the data and reducing the influence of unbalanced datasets. In general, data enhancement was attempted to tackle the bias of learned data distribution.

Acknowledgement

This work was supported by National Natura Science Foundation of China (No.51675399); Shaanxi Natural Science Foundation General Program (No.2021JM-359); and Yulin Industry-University-Research Cooperation

Project (No.2019-172).

Declaration of conflicting interests

The authors have no conflict of interests related to this publication.

References

[1] VENKATASUBRAMANIAN V, RENGASWAMY R, KEWEN Y, et al. A review of process fault detection and diagnosis. Computers & Chemical Engineering, 2003, 27(3): 293-311.

[2] ZHANG X Y, LUAN Z Q, LIU X L. A review of rolling bearing fault diagnosis based on deep learning. Equipment Management and Mai. 2017, 18: 130-133.

[3] LEI Y G, JIA F, KONG D T, et al. Opportunities and challenges of machinery intelligent fault diagnosis in big data era. Journal of Mechanical Engineering, 2018, 54(5): 94-104.

[4] COSTILLA-REYES O, SCULLY P, OZANYAN K B. Deep neural networks for learning spatio-temporal features from tomography sensors. IEEE Transactions on Industrial Electronics, 2018, 65(1): 645-653.

[5] JIN W O, JEONG J. Convolutional neural network and 2-d image based fault diagnosis of bearing without retraining// The 3rd International Conference on Compute and Data Analysis, Kahului HI USA March 14-17, 2019, Kahului, USA. New York: Association for Computing Machinery, 2019: 135-139.

[6] ZHAO C, SUN J L, LIN S L, et al. Fault diagnosis method for rolling mill multi row bearings based on AMVMD-MC1DCNN under unbalanced dataset. Sensors, 2021, 21(16): 5494.

[7] HAN T, LIU C, YANG W, et al. A novel adversarial learning framework in deep convolutional neural network for intelligent diagnosis of mechanical faults. Knowledge-Based Systems, 2019, 165: 474-487.

[8] HAO S, GE F X, LI Y, et al. Multisensor bearing fault diagnosis based on one-dimensional convolutional long short-term memory networks. Measurement, 2020, 159: 107802.

[9] GUO L, LEI Y, XING S, et al. Deep convolutional transfer learning network: a new method for intelligent fault diagnosis of machines with unlabeled data. IEEE Transactions on Industrial Electronics, 2019, 66(9): 7316-7325.

[10] MAO W T, LIU Y M, DING L, et al. Imbalanced fault diagnosis of rolling bearing based on generative adversarial network: a comparative study. IEEE Access, 2019, 7: 9515-9530.

[11] XUE Z Z, MAN J F, PENG C, et al. Research on bearing fault diagnosis based on WGAN and GAPCNN under imbalance of data. Journal of Computer Applications, 2020, 37(12): 3681-3685.

[12] LI F, ZHANG X, ZHANG X, et al. Cost-sensitive and hybrid-attribute measure multi-decision tree over imbalanced data sets. Information Sciences, 2018, 422: 242-256.

[13] CORDON I, GARCIA S, FERNANDEZ A, et al. Imbalance: Oversampling algorithms for imbalanced classification in R. Knowledge-Based Systems, 2018, 161: 329-341.

[14] GOODFELLOW I J, POUGET-ABADIE J, MIRZA M, et al. Generative adversarial networks. Communications of the ACM, 2020, 63(11): 139-144.

[15] LEE Y O, JO J, HWANG J. Application of deep neural network and generative adversarial network to industrial maintenance: A case study of induction motor fault detection//2017 IEEE International Conference on Big Data (Big Data), December 11-14, 2017, Boston, MA, USA. New York: IEEE, 2017: 3248-3253.

[16] WANG Z, WANG J, WANG Y. An intelligent diagnosis scheme based on generative adversarial learning deep neural networks and its application to planetary gearbox fault pattern recognition. Neurocomputing, 2018, 310: 213-222.

[17] HUANG N E, SHEN Z, LONG S R, et al. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. Proceedings of the Royal Society of London Series A, 1998, 454(1971): 903-998.

[18] WU Z H, HUANG N E. Ensemble empirical mode decomposition: a noise-assisted data analysis method. Advances in Adaptive Data Analysis, 2009, 1(1): 1-41.

[19] VALLENDER S S. Calculation of the Wasserstein distance between probability distributions on the line. Theory of Probability & Its Applications, 1974, 18(4): 784-786.

[20] JIANG R, PACCHIANO A, STEPLETON T, et al. Wasserstein fair classification//The 35th Uncertainty in Artificial Intelligence, July 28, 2019, Silvia Chiappa. PMLR, 2019: 862-872.

[21] DOSOVITSKIY A, BROX T. Generating images with perceptual similarity metrics based on deep networks//The 30th Conference on Neural Information Processing Systems, December 5-10, Barcelona, Spain. California: NIPS, 2016: 658-666:.

[22] MING X, KANG D. Corrections for frequency, amplitude and phase in a fast Fourier transform of a harmonic signal. Mechanical Systems and Signal Processing, 1996, 10(2): 211-221.

基于双向生成对抗网络的滚动轴承智能诊断方法

张　皓，谷立臣*，郭子辰

西安建筑科技大学 机电工程学院，陕西 西安 710055

摘　要：滚动轴承是旋转机械中的关键部件，直接影响设备的可靠性。人工智能的发展在轴承故障诊断领域取得了令人瞩目的成就。然而，滚动轴承数据集的不平衡（正常样本远丰富于故障样本）会导致诊断模型精度较低。为了解决这个问题，本文提出了一种基于双向生成对抗网络（BiGAN）的故障诊断方法。首先，通过集合经验模式分解对信号进行去噪，使其自动分配到一个合适的参考尺度，并避免模态混叠。其次，构建含有梯度惩罚项的BiGAN模型，利用单样本离差标准化方法稳定模型训练过程，实现故障样本扩充。最后，基于增强的训练集建立具有批归一化、最大池化层的卷积神经网络（CNN）诊断模型。实验结果表明，该方法提高了故障诊断的准确性和鲁棒性。

关键词：滚动轴承；故障诊断；双向生成对抗网络（BiGAN）；卷积神经网络（CNN）；数据不平衡

引用格式：ZHANG Hao, GU Lichen, GUO Zichen. Intelligent diagnosis method of rolling bearing based on BiGAN. Journal of Measurement Science and Instrumentation, 2024, 15(2): 264-275.