IOP Publishing Measurement Science and Technology
Meas. Sci. Technol. **31** (2020) 055009 (18pp) https://doi.org/10.1088/1361-6501/ab50f0

Rolling bearing fault diagnosis based on intelligent optimized self-adaptive deep belief network

Shuzhi Gao[1], Lintao Xu[2], Yimin Zhang[1] and Zhiming Pei[2]

[1] Equipment Reliability Institute, Shenyang University of Chemical Technology, Shenyang 110142, People's Republic of China
[2] College of Information Engineering, Shenyang University of Chemical Technology, Shenyang 110142, People's Republic of China

E-mail: zhangyimin_126163@126.com

Received 26 April 2019, revised 14 August 2019
Accepted for publication 24 October 2019
Published 31 January 2020

Abstract

Due to the structure of the rolling bearing itself and the complexity of its operating environment, the collected vibration signals tend to show strong non-stationary and time-varying characteristics. It has been a challenge to extract useful fault signature information from actual bearing vibration signals and identify bearing faults in the field of machinery in recent years. Therefore, this paper proposes a novel optimized self-adaptive deep belief network (DBN). The DBN is pre-trained by a minimum batch stochastic gradient descent, and then a back-propagation neural network and conjugate gradient descent are used to supervise and fine-tune the entire DBN model, which effectively improves the classification accuracy of the DBN. A salp swarm algorithm is used to optimize the DBN, and then the experience of the DBN structure is summarized. The vibration signal of the rolling bearing is analyzed by this method and it is confirmed that it has higher diagnostic accuracy and better convergence.

Keywords: deep belief network, salp swarm algorithm, rolling bearing fault diagnosis, optimization design

(Some figures may appear in colour only in the online journal)

1. Introduction

Rolling bearings are the most common mechanical components and play an extremely important role in almost all rotating machinery. Due to their relatively low price and easy operation, they are widely used in industry. The maneuvering of the rotating machine is completely dependent on the health of the rolling bearing, which accounts for approximately 45%–55% of these mechanical equipment failures. The causes of bearing failure, such as wear, pitting, dust, lubricant contamination, temperature change and excessive load, can lead to the catastrophic collapse of the entire system and decrease the reliability and availability of equipment. Since the vibration signal is directly related to the rolling bearing structure, some methods based on signal processing have unique advantages in this respect. The vibration signal generated by the bearing fault is a kind of non-stationary signal containing various interference noise. How to extract the feature information of the bearing fault from the non-stationary signal is key to identifying the bearing fault. Accurate diagnosis and identification of rolling bearing faults can guarantee the normal operation of mechanical equipment.

The bearing vibration signal processing method is usually divided into two steps: extracting features from the signal and making decisions on those features. The vibration response of a defective bearing consists of a series of pulses which are generated when the rolling element runs through the bearing defect surface. The frequency generated by the pulse is called the bearing characteristic frequency, which depends on the shaft speed, bearing geometry and defect

position [1–3]. Generally speaking, there are four fault frequencies for ball bearings: ball pass frequency on outer race, ball pass frequency on inner race, ball spin frequency and fundamental train frequency, which correspond to the defects of the outer ring, the inner ring, the rolling elements and the cage, respectively. Nowadays, fault diagnosis of rolling bearings plays a very important role in the field of mechanical research [4]. Samanta and Nataraj [5] studied the application of particle swarm optimization (PSO) computational intelligence technology in machine bearing fault detection, and then used a PSO algorithm to optimize the input parameters in artificial neural network (ANN) and support vector machine (SVM) classifiers, respectively. The vibration signals of normal and fault conditions were mainly subjected to feature extraction and classification tests. Lin and Wang [6] proposed a novel method for fault diagnosis of rolling bearings—the wavelet neural network optimized by simulated annealing particle swarm optimization was used to identify the faults of rolling bearings. Song *et al* [7] proposed a new method for fault diagnosis of roller bearings based on the fusion of hierarchical entropy and PSO-SVM. This method can effectively and accurately identify the fault class and fault severity of the roller bearing. Rajeswari *et al* [8] proposed a novel multi-class SVM classification method based on the vibration signal for rolling bearing state prediction. The proposed method promoted the classification accuracy of SVM. Chen *et al* [9] proposed an intelligent diagnosis model for rolling bearing faults based on a multi-core SVM and chaotic particle swarm optimization, which improved the diagnosis accuracy of bearing faults. Zhao *et al* [10] proposed a method of bearing fault diagnosis based on an improved frog leaping algorithm combined with the back-propagation (BP) neural network. The model established by this method had better generalization ability and robustness. Liang *et al* [11] proposed the ensemble local characteristic-scale decomposition (ELCD) and extreme learning machine (ELM) method for rolling bearing fault diagnosis, where ELCD was used to decompose the vibration signal, the sensitivity characteristics were obtained, and finally it was verified that ELCD-ELM could effectively identify bearing faults. Fu *et al* [12] proposed a hybrid optimization algorithm combining mutation operator, grey wolf optimizer (GWO), and sine cosine algorithm (SCA), termed mutation hybrid GWO-SCA (MHGWOSCA)-optimized SVM model, which was used to classify different fault samples and showed good accuracy and stability in all classification indicators. Wei *et al* [13] proposed a method-optimal variational mode decomposition (VMD), and the envelope entropy was used as a fitness function of whale optimization algorithm (WOA), meanwhile, which was applied to optimizing VMD and searching optimal parameters. Finally, envelope demodulation analysis was applied to process intrinsic mode function (IMF) components, and it was proved that this method had obvious advantages in extracting fault features of the rolling bearing. Ma *et al* [14] proposed a transfer learning convolutional neural network based on AlexNet, which converted the traditional 1D vibration signal feature extraction issue to a 2D time-frequency image processing problem, and realized a better bearing classification effect; however, the training process was time-consuming, which also provided a research basis for future deep learning technology application.

The ability of shallow learning methods, such as BP and SVM, is mainly determined by the extracted feature quality. Some features that are liable to be identified are manually selected, which depends largely on researchers' experience. Furthermore, the same features may not necessarily be suitable for new problems [15]. According to several studies, the deep learning architecture has better representation ability than the shallow learning model. Therefore, it is extremely significant that a deep learning model will be established for automatic learning and diagnosis of features [16]. Hinton proposed a new deep network model—the deep belief network (DBN)—which can make the entire neural network generate training data with maximum probability by training the weights between hidden layers. The DBN can identify features, and classify and generate data [17, 18]. Huang *et al* proposed a novel DBN model and then applied it to the field of unsupervised feature learning, image processing and fault diagnosis, which finally verified its learning performance [19]. To solve the problem of the diversity and complexity of the actual vibration signal of the rolling bearing, Shao *et al* [20] proposed a PSO-DBN method for bearing fault diagnosis; however, the extraction and classification of time-frequency domain features were not considered in this work. Shao *et al* [21, 22] proposed an adaptive DBN method based on double tree complex wavelet packet transform and a modified convolution DBN model based on compressed sensing to improve the fault diagnosis effect of the rolling bearing, but there were no explanations and demonstrations for the confirmation of structural parameters of neural networks in the above two studies.

According to the abovementioned research results, the majority of improved DBN methods proposed by scholars only focused on the calculation of learning rate and extracted features, and there are few studies on the neural network structure and momentum which have a great effect on DBN performance. The major contributions in this paper are as follows.

(1) In order to quickly achieve higher classification accuracy results, salp swarm algorithm (SSA) optimization is used to optimize the network structure of the DBN, and the construction experience of the deep learning neural network will be summarized.
(2) By constructing a self-adaptive restricted Boltzmann machine (SARBM), the convergence effect and classification accuracy of the DBN itself are further improved.

2. DBN theory and SSA

2.1. DBN

Deep learning has become one of the most popular research hotspots in the field of machine learning. As a new method in the field of machine learning, deep learning has shown its strong nonlinear fitting ability in image, text and voice signals. An important point of deep learning is that it can automatically extract features. It can be divided into two categories: unsupervised learning and supervised learning. Different

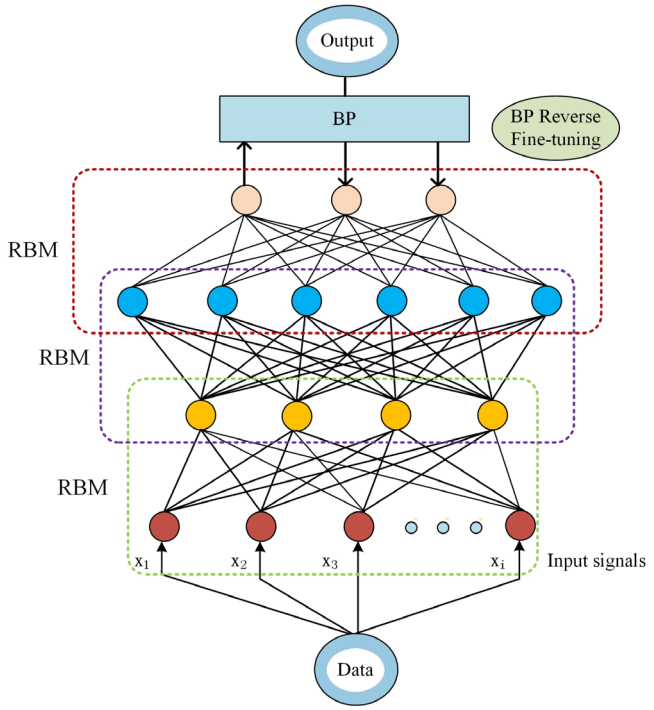

Figure 1. Overall structure of the DBN with three RBMs.

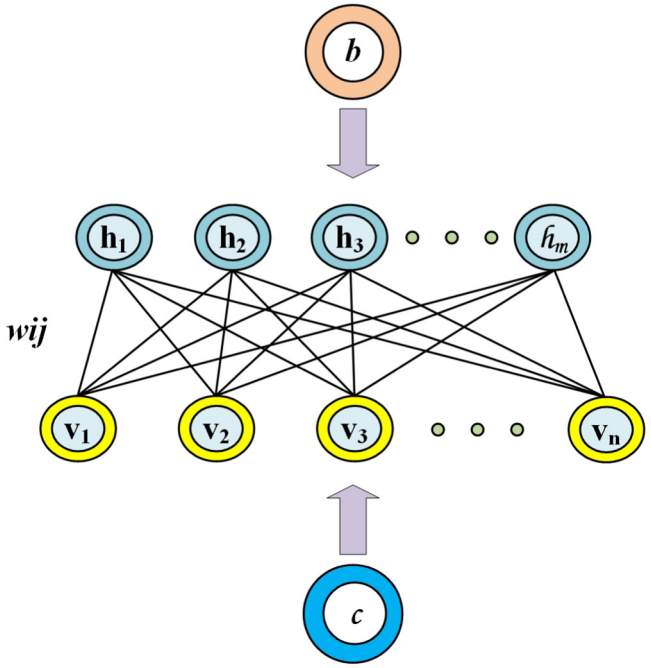

Figure 2. RBM network structure.

learning methods can construct different machine learning models, such as k-nearest neighbor (KNN), decision tree and convolutional neural network (CNN), which are deep supervised learning models. The DBN used in this paper is an unsupervised learning model.

The DBN is a probability generation model. The parameters between layers are updated by training to improve the accuracy of neural network output. The DBN can also be seen as a deep BP neural network, which is a deep neural network architecture stacked by a multi-layer restricted Boltzmann machine (RBM). In this paper, DBN uses the unsupervised feature learning method to learn the effective characteristics of vibration signals. This method has been verified in the field of image and voice signals [23].

Generally speaking, the training process of DBN is divided into two steps: pre-training and fine-tuning. In the pre-training phase, an unsupervised greedy layer-by-layer learning algorithm is adopted as follows.

Initialize the network parameters and input the extracted samples into the visual layer of the RBM. The output value of the hidden layer is treated as training data and input to the next RBM. Repeat the above process multiple times, finally output the optimal value, and achieve the purpose of deep mining the essential features of the data. The fine-tuning phase uses the BP neural network algorithm to perform supervised reverse adjustment of the entire DBN weight parameter. Figure 1 shows the overall structure of the DBN.

2.2. RBM

The RBM is the basic structure of the DBN, which has a two-layer network structure. One layer is the visible layer, which is usually called the input layer, and the other layer is the hidden layer, which is also called the feature extraction layer. The connection between the visible layer and the neurons of the hidden layer is a bidirectional full connection. The RBM network structure is shown in figure 2.

In the RBM, the weight w between any two neurons denotes the connection strength, c denotes the visible-layer bias coefficient, and b denotes the hidden-layer bias coefficient. $v = (v_1, v_2 \cdots, v_n)$ represents the state of neurons in the visible layer, and $h = (h_1, h_2 \cdots h_m)$ represents the state of hidden-layer neurons. Actually, the abovementioned h and v represent the weights of neurons in the hidden and visible layers, respectively, where v is the input vector and h is the output vector. Since RBM is essentially a probabilistic model based on energy, the energy function E of the visible unit and the hidden unit can be defined as follows:

$$E(v, h|\theta) = -\sum_{i=1}^{n} c_i v_i - \sum_{j=1}^{m} b_j h_j - \sum_{i=1}^{n}\sum_{j=1}^{m} v_i w_{ij} h_j \quad (1)$$

where w_{ij} is the weight of visible-layer neurons i connected with hidden-layer neurons j, and n and m are the number of visible-layer neurons and hidden-layer neurons, respectively. $\theta = (w_{ij}, c_i, b_j)$ is the set of multiple model parameters.

The energy function of the RBM's joint probability distribution is as follows:

$$P(v, h|\theta) = \frac{1}{L(\theta)} e^{-E(v,h|\theta)} \quad (2)$$

where $L(\theta)$ is a normalized factor, also known as the partition function. The conditional probabilities of the visible and hidden layers are obtained by equations (3) and (4), respectively:

$$P(h|v;\theta) = \prod_j P(h_j|v;\theta) \quad (3)$$

Table 1. Procedure of the optimization method (SSA).

Step	Concrete contents
Step1	Import the training sample set into the DBN.
Step2	Initialize a series of DBN parameters (population total, learning rate, etc).
Step3	Initialize each salp position within the given range, the initial speed $v_0 = 0$.
Step4	The cumulative reconstruction error (fitness value) between the training sample and the misclassified sample model output is used as a comparison condition. After several iterations, the optimal individual position (food source position) is selected.
Step5	The salp position is updated by equations (12), (13) and (15).
Step6	The fitness value of the final output should be less than a certain threshold (experience value), and then output the result, otherwise it is executed until the number of iterations reaches the set value.

$$P(v|h;\theta) = \prod_i P(v_i|h;\theta). \tag{4}$$

In this paper, the standard binary RBM is replaced by Gaussian GRBM. The hidden-layer neurons in GRBM obey Gaussian distribution, while the visible-layer neurons, similar to the standard RBM, still follow Bernoulli distribution.

The energy function E_g of the Gauss–Bernoulli RBM (GBRBM) distribution is as follows:

$$E_g(v,h;\theta) = \sum_{i=1}^{n}\frac{(v_i - b_i)^2}{2\delta_i^2} - \sum_{i=1}^{n}\sum_{j=1}^{m} w_{ij}\frac{v_i}{\delta_i}h_j - \sum_{j=1}^{m} c_j h_j \tag{5}$$

where δ_i is the standard deviation of the training samples, the activation function of the hidden layer neurons is still *sigmoid*, and the activation function of the layered neurons becomes a Gaussian function. The conditional probability distribution is defined by equation (6):

$$P_g(v_i|h_j) = N(v_i|\mu_i,\delta_i^2) = \frac{1}{\delta_i\sqrt{2\pi}}exp[-\frac{1}{2\delta_i^2}(v_i - b_i - \delta_i\sum_j w_{ij}h_j)^2]. \tag{6}$$

The mean μ_i of the Gaussian distribution is as follows:

$$\mu_i = c_i + \delta_i^2\sum_j w_{ij}h_j. \tag{7}$$

In general, standard deviation $\delta_i = 1$. For GRBM visible-layer neurons, the learning rate is one to two orders of magnitude smaller than the binary value.

The abovementioned parameter θ can be obtained by maximizing the logarithmic likelihood function in the RBM training set. If the number of training samples is T, the maximum likelihood function F is as follows:

$$\theta^* = \text{argmax}F(\theta) = \frac{1}{T}\text{argmax}\sum_{t=1}^{T}\log P(v^t|\theta). \tag{8}$$

The maximum gradient of $F(\theta)$ and the optimal parameter value are obtained by stochastic gradient descent (SGD). Each RBM needs to be repeated several times. However, these parameters have a different update orientation after each iteration, which may cause early convergence or instability of the algorithm. Therefore, it is necessary to add a momentum factor to solve this problem when the parameter is updated. The update rule is as follows:

$$\Delta w_{ij}^{(k)} = \varepsilon\frac{\partial F(\theta)}{\partial w_{ij}^{(k)}} + \eta\Delta w_{ij}^{(k-1)} \tag{9}$$

$$\Delta c_i^{(k)} = \varepsilon\frac{\partial F(\theta)}{\partial c_i^{(k)}} + \eta\Delta c_i^{(k-1)} \tag{10}$$

$$\Delta b_j^{(k)} = \varepsilon\frac{\partial F(\theta)}{\partial b_j^{(k)}} + \eta\Delta b_j^{(k-1)} \tag{11}$$

where η is the learning rate, k is the number of iterations, and ε is the momentum factor. The introduction of the momentum factor has better anti-oscillation ability for the parameters in the process of the training RBM model. In this paper, the structure of the DBN consists of three RBMs: the first one is GRBM, which mainly converts the input samples into binary values, the second and third are GBRBMs [24], then further dealing with the input data.

2.3. SSA

Mirjalili *et al* created the mathematical model of the salp chain for solving single-objective and multi-objective optimization problems [25]. The idea was to divide the population into two groups: leaders and followers. In other words, a parent led a group of offspring, and the descendant population follows each other. Researchers demonstrated that this structure can help the sea squirt to quickly coordinate movement and foraging.

The total population size of the SSA is N. The individual location is defined in the D-dimensional search space, which is the number of variables of the objective function. The upper and lower bounds of each variable $ub = [ub_1, ub_2, \cdots, ub_D]$ and $lb = [lb_1, lb_2, \cdots, lb_D]$, respectively. All individual location sets are stored in matrix X. The position of the food source is set to f, which will be used as a search target for the entire group of salps. The leader then searches for neighboring locations of the food source. The location update rules are defined by equations (12) and (13):

$$X_j^1 = f_j - c_1(ub_j - lb_j)\cdot c2 + lb_j, c_3 < 0 \tag{12}$$

$$X_j^1 = f_j + c_1(ub_j - lb_j)\cdot c2 + lb_j, c_3 \geqslant 0 \tag{13}$$

where X_j^1 is the first salp (leader) of the jth dimension, f_j is the food source location of the jth dimension, ub_j is the upper bound of the jth dimension, ub_j is the lower bound of the jth dimension, and c_1, c_2 and c_3 are random numbers. c_1 has a great influence on the effect of the SSA, and enhances each

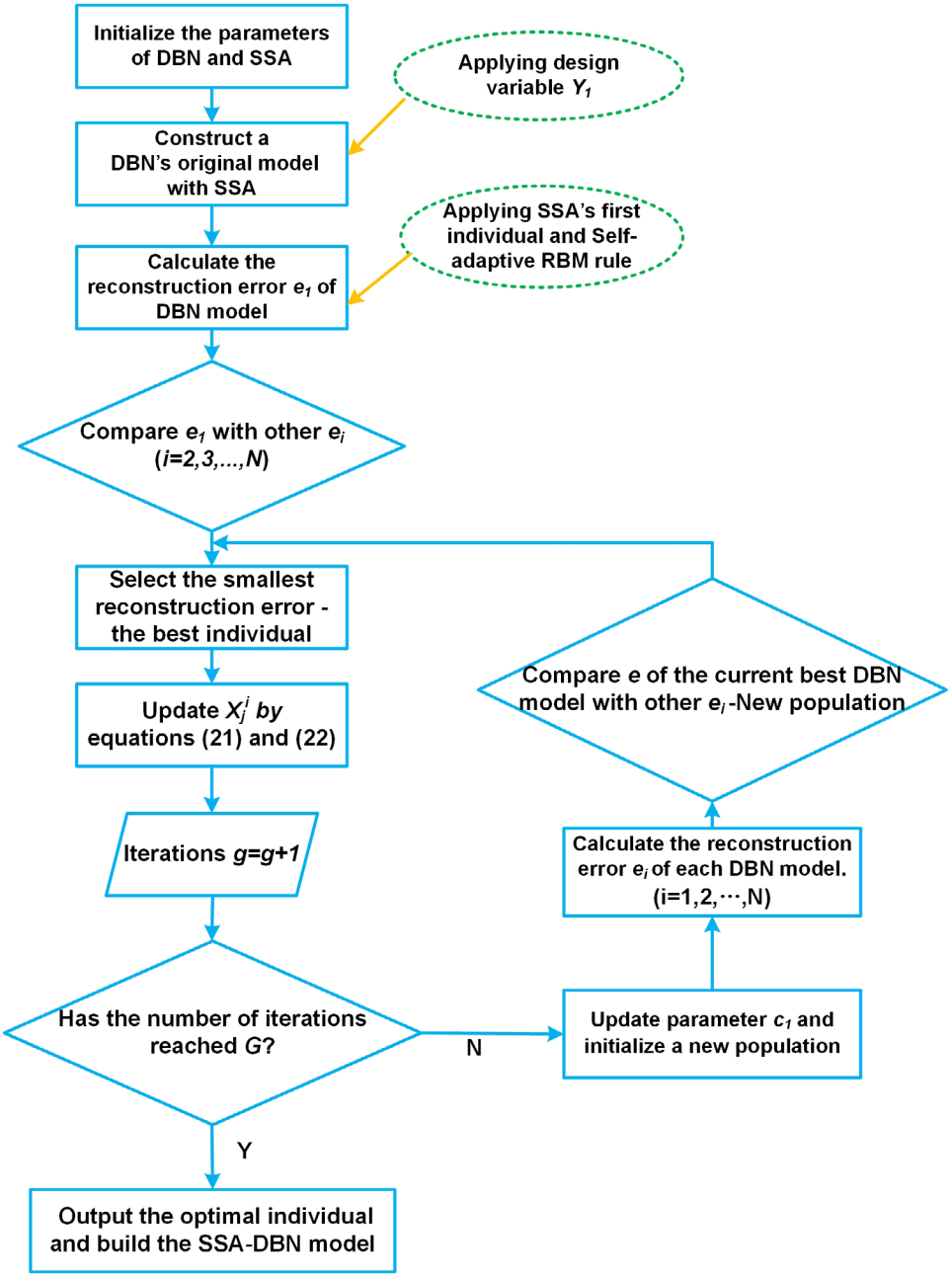

Figure 3. Flow chart of SSA-DBN.

individual's ability to explore the whole space in the early and late stage of the search. c_1 can be defined by equation (14):

$$c_1 = 2e^{-(\frac{4g}{G})^2} \tag{14}$$

where g is the current number of iterations, and G is the maximum number of iterations. Both c_2 and c_3 obey Gaussian distribution, and both are random numbers in the range [0,1].

The location update rule for the remaining individuals in SSA (Newton's law of motion) is as follows:

$$X_j^i = \frac{1}{2}a_s t_s^2 + v_0 t \tag{15}$$

where $i \geqslant 2$, t_s is the magnitude of time, v_0 is the initial velocity, a_s is the acceleration, and $a_s = \frac{v_{\text{final}} - v_0}{t}$, $v_{\text{final}} = \frac{(X_j^{i-1} - X_j^i)}{t}$, t is the number of iterations, $t = 1$, $v_0 = 0$.

Therefore, the equation (15) can be expressed as

$$X_j^i = \frac{1}{2}(X_j^i + X_j^{i-1}). \tag{16}$$

The main idea of the SSA is as follows: firstly, the individual position of the population is initialized, and then the calculated optimal fitness value is used to determine the optimal individual. Secondly, the obtained optimal individual is taken

Table 2. Extracted time domain feature parameters.

Time domain feature parameters	Description
1. Variance	$F_{t1} = \sum_{s=1}^{T}(x_s - \bar{x})^2 T^{-1}$
2. Standard deviation	$F_{t2} = \sqrt{\sum_{s=1}^{T}(x_s - \bar{x})^2 T^{-1}}$
3. Mean square root	$F_{t3} = \sqrt{\sum_{s=1}^{T} x_s^2 T^{-1}}$
4. Skewness	$F_{t4} = \sum_{s=1}^{T}(x_s - \bar{x})^3 (T-1)^{-1}\Psi^3$
5. Kurtosis	$F_{t5} = \sum_{s=1}^{T}(x_s - \bar{x})^4 (N-1)^{-1}\Psi^2$
6. Crest indicator	$F_{t6} = \max\lvert x_s\rvert((\sum_{s=1}^{T} x_s^2)T^{-1})^{-\frac{1}{2}}$
7. Clearance indicator	$F_{t7} = \max\lvert x_s\rvert((\sum_{s=1}^{T} \sqrt{\lvert x_s\rvert})T^{-1})^{-2}$
8. Impulse indicator	$F_{t8} = \max\lvert x_s\rvert((\sum_{s=1}^{T} \lvert x_s\rvert)T^{-1})^{-1}$
9. Peak indicator	$F_{t9} = \sqrt{T\sum_{s=1}^{T}(x_s)^2}(\sum_{s=1}^{T}\lvert x_s\rvert)^{-1}$

x_s is the bearing vibration signal sequence, $\bar{x}$ is the signal mean, and T is the total number of data samples.

Table 3. Extracted frequency domain feature parameters.

Frequency domain feature parameters	Description
1. Mean frequency	$F_{1f} = \frac{1}{M}\sum_{m=1}^{M}\mu(m)$
2. Center frequency	$F_{2f} = \sum_{m=1}^{M}(f_m\mu(m))(\sum_{m=1}^{M}\mu(m))^{-1}$
3. Root mean square frequency	$F_{3f} = \sqrt{\sum_{m=1}^{M}(f_m^2\mu(m))\sum_{m=1}^{M}\mu(m)}$
4. Standard deviation frequency	$F_{4f} = \sqrt{(f_m - F_2)^2\mu(m)M^{-1}}$
5. Kurtosis frequency	$F_{5f} = ((f_m - F_2)^4\mu(m)M^{-1})F_4^4$

$\mu(m)$ is the spectrum, M is the number of spectral lines, and f_m is the frequency of the mth line.

as the leader, and its individual position is taken as the food source position. It is equivalent to recording the best position for each food source during the iteration, which means that the followers do not easily fall into the local optimum and thus improves the convergence of this algorithm.

3. The proposed method (SSA-DBN)

The performance of DBN performance mainly depends on the construction of the RBM, the structure of the DBN itself and the selection of various parameter factors, which affect the convergence and classification effect of DBN.

3.1. Optimization design of DBN

The structural characteristics of the neural network are the number of hidden layers and the number of cells per layer. These characteristic parameters have a great influence on the performance of the neural network. That is to say, in order to obtain suitable attributes of data classification or prediction, it is necessary to find an optimal feature parameter. However, in theory, there is no advanced technology to determine the

Table 4. Extracted time-frequency domain feature parameters.

Time-frequency domain feature parameters	Description
1. EMD energy spectrum	$F_{1tf} = -\sum_{k=1}^{K} p_k \log_{10} p_k$
2. LMD energy spectrum	$F_{2tf} = E_j = \int_{-\infty}^{+\infty}\lvert PF_j(t)\rvert^2 \mathrm{d}t$
3. Wavelet packet energy entropy	$F_{3tf} = -\sum_{s=1}^{T}\zeta_{I,n}(s), (n = 0, 1, \cdots 2^I - 1)$

p_k is the ratio of the kth IMF component to the total energy, and K is the number of IMF components. $j = 1, 2, \cdots J$, there are a total of J PF components, and E_i is the total energy of the ith PF component. The signal energy distribution of the sequence of decomposition coefficients of the nth node of the I layer.

structure of each hidden layer, so choosing the number of optimal hidden-layer neurons is the main problem now. In this paper, three RBMs are selected as the basic constituent elements of the DBN. SSA is an intelligent optimization algorithm proposed in the past two years. Although it has few application fields at this stage, the algorithm has great advantages in high-dimensional problems.

In this paper, the optimal DBN structure is designed by using the number of neurons in each hidden layer which are used as optimization design variables. The determination of the number of hidden-layer nodes is an important part of the network design, because the number of nodes is directly related to the requirements of solving the problem and the number of input and output units. Moreover, if the number of hidden-layer nodes is too small, the number of connection weight combinations generated is not sufficient to satisfy the learning of several samples. However, the excessive number of hidden-layer nodes may result in poor generalization of neural networks. The number of neurons in the three RBMs is set to z_1, z_2, z_3, respectively. The learning rate $\eta \in (0, 1)$, and the momentum $\varepsilon \in [0, 1)$. Each individual $Y_i(z_1, z_2, z_3)$ in SSA is set to a three-dimensional vector. The total number of particles is N. The structural parameters of the DBN are improved by the SSA and applied to the experimental research in this paper.

According to the literature [26], the range of the upper and lower bounds of the initial individuals z_1, z_2, z_3 in the SSA is (0,500). However, in the process of studying the structure of neural networks, we found that there is a numerical progressive relationship from the number of neurons in the first hidden layer to the number of neurons in the last hidden layer, which affects the classification performance of the DBN to a large extent. In the fourth chapter of the experimental study, the accuracy of the relationship will be confirmed.

This relationship can be defined by equations (17) and (18):

$$z_2 = z_1 \times \frac{1}{2} \pm \mathrm{rand} \times (1, 20), (200 \leqslant z1 \leqslant 400) \quad (17)$$

$$z_3 = z_1 \times \frac{1}{3} \pm \mathrm{rand} \times (1, 20), (200 \leqslant z1 \leqslant 400). \quad (18)$$

Figure 4. Flow chart of bearing fault diagnosis.

The number of neurons in each hidden layer is determined by the above relationship in turn. The best structure of the DBN will be found quickly by this method.

The SSA determines the optimal structure and parameters of the DBN. Table 1 shows the concrete steps of the optimization method.

3.2. Structure of SADBN

The learning rate η and the momentum factor ε in the RBM are extremely important parameters that affect the convergence speed and classification performance of the DBN. At this stage, researchers have made a number of improvements to the learning rate, but there are very few studies on momentum.

Table 5. Procedure of the proposed method for bearing fault diagnosis.

Step	Description
Step1	Input the bearing data and understand each class of bearing condition.
Step2	Extract the time domain, frequency domain and time-frequency domain features of all data samples.
Step3	Initialize the parameters of SADBN, input the extracted features into the deep network.
Step4	Optimize the structure parameters of SADBN using SSA.
Step5	Train the first layer of improved RBM in the deep network model, and apply the minimum batch stochastic gradient descent to the maximum likelihood function for gradient descent in the RBM. The output value is entered into the next RBM until iterating to the last RBM.
Step6	The conjugate gradient descent and BP neural network are used to supervise and fine-tune the entire DBN model, so that the model convergence effect will be better.
Step7	Then, input the test data into the SSA-DBN model to diagnose classification of test data.
Step8	Classification accuracy of various bearing conditions is obtained, and the final results are counted.

In the standard DBN training process, the ε is given a determined empirical value, which will reduce the adaptability and convergence of the entire neural network parameters.

Therefore, this paper proposes an adaptive strategy to improve the momentum factor. Similar to the standard RBM, at each iteration, the cumulative reconstruction error before and after the iteration is used as the evaluation criterion—the standard cumulative reconstruction error (loss function). It is defined by equation (19):

$$e(i) = \frac{1}{2}\sum_{q=1}^{Q}\sum_{j=1}^{\text{epoch}}\left(V_{i,j}^{\text{data}} - V_{i,j}^{\text{rec}}\right) \tag{19}$$

where q is the current number of iterations of the RBM, Q is the maximum number of iterations of the RBM, epoch is the total number of batches of input data, $V_{i,j}^{\text{data}}$ is the original input data of the jth batch of the ith generation, and $V_{i,j}^{\text{rec}}$ is the reconstructed state of the input data of the jth batch of the ith generation.

The adaptive momentum strategy is described as follows:

The initial momentum factor $\varepsilon = 0.5$, and the threshold factor $\rho = 0.01$. When the cumulative reconstruction error decreases with the number of iterations, the momentum factor will increase to some extent. When the cumulative reconstruction error increases gradually, the momentum factor will gradually decrease. The pseudo code for this adaptive strategy is as follows:

$\Delta e_q = e(q) - e(q-1), (q \geqslant 2)$
If $\Delta e_q > \rho$, *then* $\varepsilon^q = p_d\varepsilon^{q-1}$
If $0 < \Delta e_q \leqslant \rho$, *then* $\varepsilon^q = p_u\varepsilon^{q-1}$
If $\Delta e_q \leqslant 0$, *then* $\varepsilon^q = \varepsilon = 0.5$

where e_q is the cumulative reconstruction error of the qth generation, p_d is the attenuation factor, which belongs to (0.2,0.5), and p_u is the increasing factor, which belongs to (0.5,0.9). This strategy is applied to the RBM to form a self-adaptive DBN (SADBN). The flow chart of the proposed method (SSA-DBN) is as shown in figure 3.

3.3. Feature extraction

In this paper, the features extracted from the input data include three kinds of features: time domain, frequency domain and time-frequency domain. SADBN has a better ability to learn some of these sensitive features and ultimately achieve higher classification accuracy. The extracted time domain features include variance, standard deviation, mean square root, skewness, kurtosis, crest indicator, clearance indicator, impulse indicator and shape indicator. The frequency domain features include mean frequency, center frequency, root mean square frequency, standard deviation frequency and kurtosis frequency. The time-frequency domain features include empirical mode decomposition (EMD) energy spectrum, local mean decomposition (LMD) energy spectrum and wavelet packet energy entropy. The formulas of all feature parameters in this paper are listed in tables 2–4, respectively.

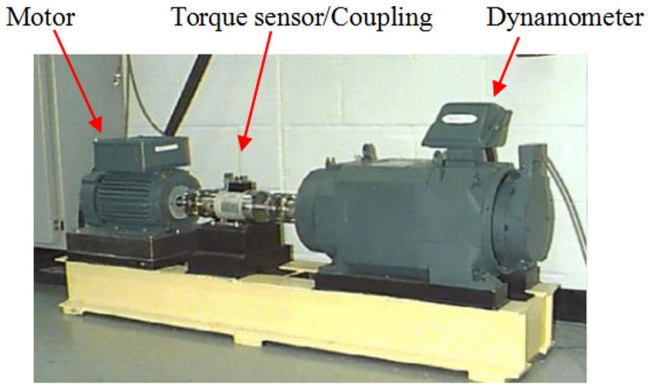

Figure 5. Rolling bearing vibration test bench.

3.4. Bearing fault diagnosis steps

This paper presents the SSA-DBN method for fault diagnosis of rolling bearings. Figure 4 shows a flow chart of the fault diagnosis of the bearing.

The specific steps of the rolling bearing fault diagnosis method proposed in this paper are shown in table 5.

4. Experiment verification

4.1. Experimental data description

Experiments are carried out to verify the effectiveness of the proposed method. Bearing vibration data obtained from the Bearing Experimental Center of Case Western Reserve University in the United States are used for analysis and

Table 6. Experimental conditions and data description.

Data categories	Bearing dataset	Load (/hp)	Motor speed (/rpm)	Sample size
First group (Dataset1)	Trainings	0	1797	1080
	Testings	1	1772	360
Second group (Dataset2)	Trainings	0	1797	1080
	Testings	2	1750	360

Table 7. Bearing experimental data demonstration.

Bearing condition	Fault size (inches)	Fault direction	Training(s):Testing(s)	Label
Inner race	0.007	/	3:1	1
Inner race	0.014	/	3:1	2
Inner race	0.021	/	3:1	3
Roller	0.007	/	3:1	4
Roller	0.014	/	3:1	5
Roller	0.021	/	3:1	6
Outer race	0.007	Center@6:00	3:1	7
Outer race	0.007	Opposite@12:00	3:1	8
Outer race	0.007	Vertical@3:00	3:1	9
Outer race	0.014	Center@6:00	3:1	10
Outer race	0.014	Opposite@12:00	3:1	11
Outer race	0.021	Center@6:00	3:1	12

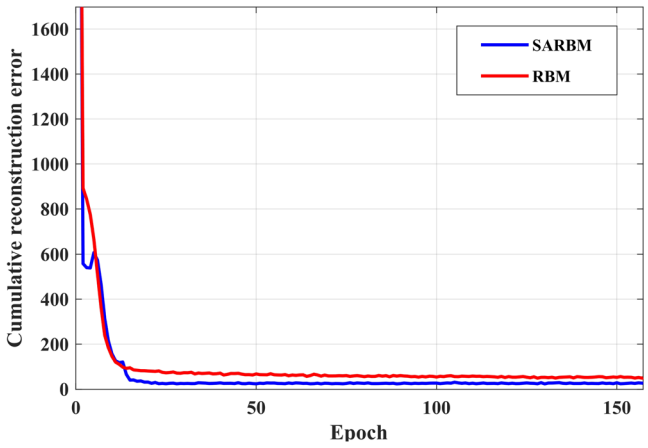

Figure 6. Test curve of SARBM and standard RBM.

verification. Figure 5 shows the vibration experiment device, which includes a 1.49 kW (2 hp) motor, a torque sensor and coupling, a dynamometer and an electronic controller (not shown in the figure). The bearing to be tested supports the rotating shaft of the motor, and the bearing of the driving end is SKF 6205. In this paper, the signal measured by the bearing driving end sensor is mainly used. This experiment uses an electro-discharge machining method for rolling bearing. The types of faults include roller failure, inner race failure and outer race failure. Its fault diameters are 0.007, 0.014 and 0.021 in (1 in = 25.4 mm), and the damage point of the bearing outer ring will be set at 3, 6 and 12 o'clock orientation. An acceleration sensor is placed above the bearing seat of the motor drive end to collect the vibration acceleration signal of the faulty bearing. The vibration signal is acquired by a 16-channel data logger with a sampling rate of 12 000 Hz. The operating power and bearing speed are measured by a torque sensor.

The experimental data set in this paper selects a total of 12 failure conditions during bearing operation, and 120 samples are taken for each failure condition. When classifying eight bearing conditions, 180 samples are taken under each working condition. The sample length of both signals is 2048 data points. The experimental data in this paper are described in detail in tables 6 and 7.

4.2. Rolling bearing intelligent fault diagnosis

The structure of the DBN is generally set to five or six layers, which is the best standard [27]. As the number of neurons in the hidden layer increases, the overall computational complexity also increases. Therefore, the number of neurons in each layer is defined by an intelligent optimization algorithm, which can be quickly found and promotes classification effectiveness.

The features of each of the following experiments are obtained by using a deep learning method, and each experiment is repeated approximately 25 times independently.

In order to verify the feature extraction ability of the SSA-DBN proposed in this paper, the research of layer-by-layer feature learning processes of the DBN in each experiment will be simulated with 2D and 3D principle component analysis (PCA) visualization techniques. It is liable to visually observe the classification effect of SSA-DBN, which further validates the powerful ability of SSA-DBN to automatically mine the data parameters from the original set of features.

4.3. Rolling bearing experiment 1

Twelve bearing conditions are selected from experiment 1 as training/testing samples, and the data selected in this experiment are from Dataset2. These 12 conditions are all single-point failures of the bearing.

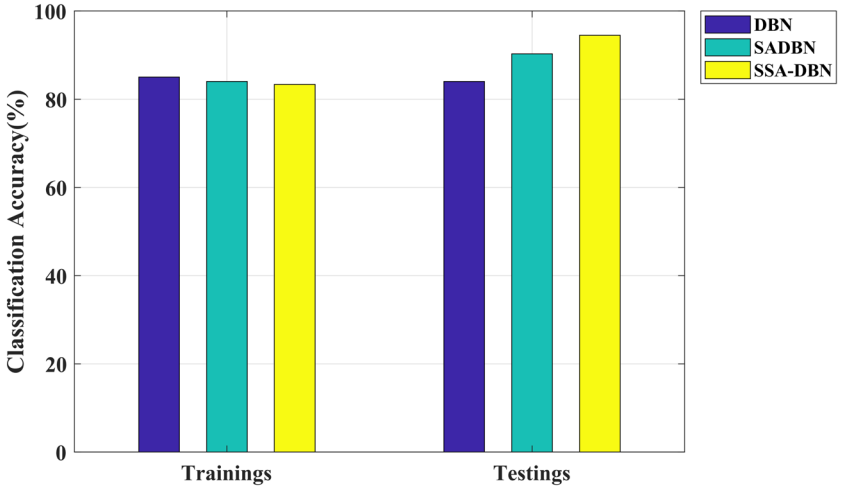

Figure 7. Comparison of the three methods in experiment 1.

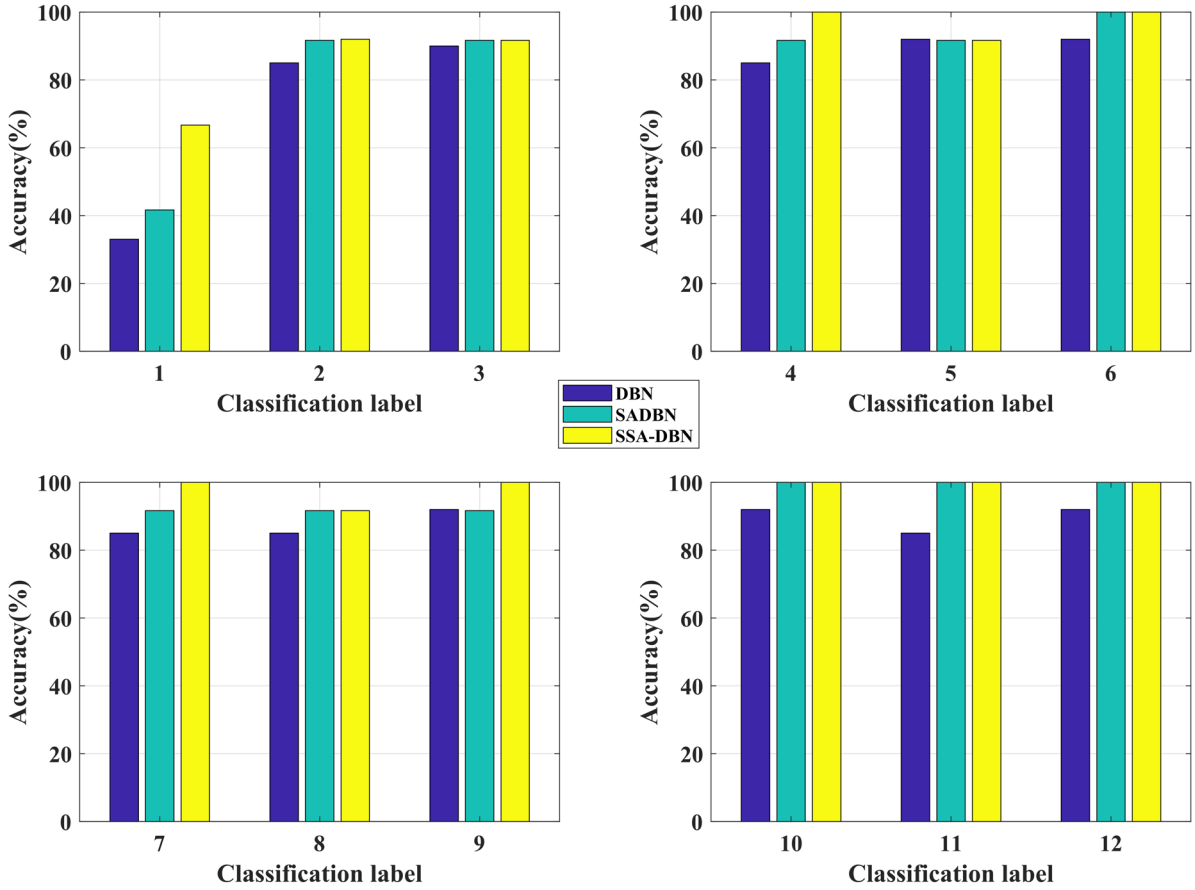

Figure 8. Classification accuracy of 12 bearing working conditions in experiment 1.

Figure 6 shows the convergence of the SARBM adaptive strategy designed in this paper, which clearly demonstrates that as the number of iterations increases, the overall convergence of SARBM is better than that of standard RBM.

The specific parameters of the experiment are as follows.

The extracted feature parameters are set to 20, which includes nine time domain parameters and 11 time-frequency domain feature parameters.

(1) **Wavelet packet transform (WPT):** The wavelet basis function is Meyer, and the number of decomposition layers is 3.
(2) **EMD:** EMD energy features of the first five IMF components.
(3) **LMD:** LMD energy spectrum features of the first five product function (PF) components.
(4) **SADBN:** The initial network structure is 20-350-260-150-12, the learning rate is 0.1, the momentum factor is

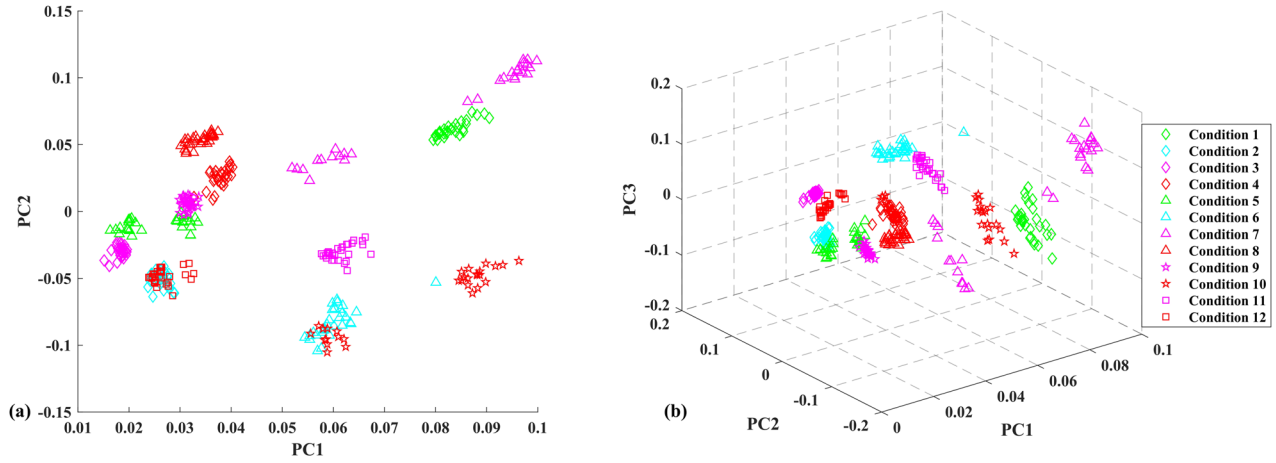

Figure 9. 2D-PCA and 3D-PCA projection of the original features in experiment 1.

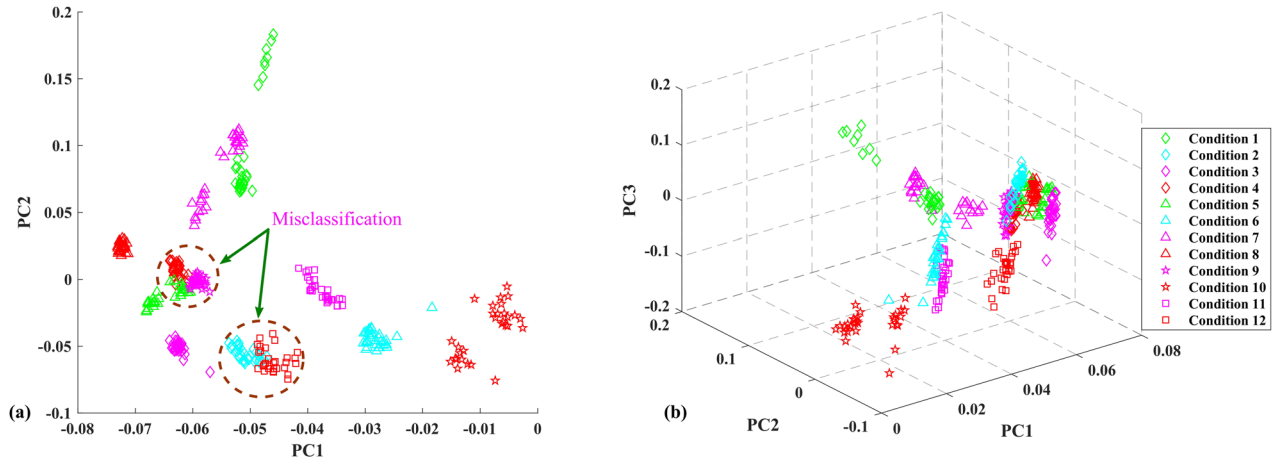

Figure 10. 2D-PCA and 3D-PCA projection of the first hidden-layer feature in experiment 1.

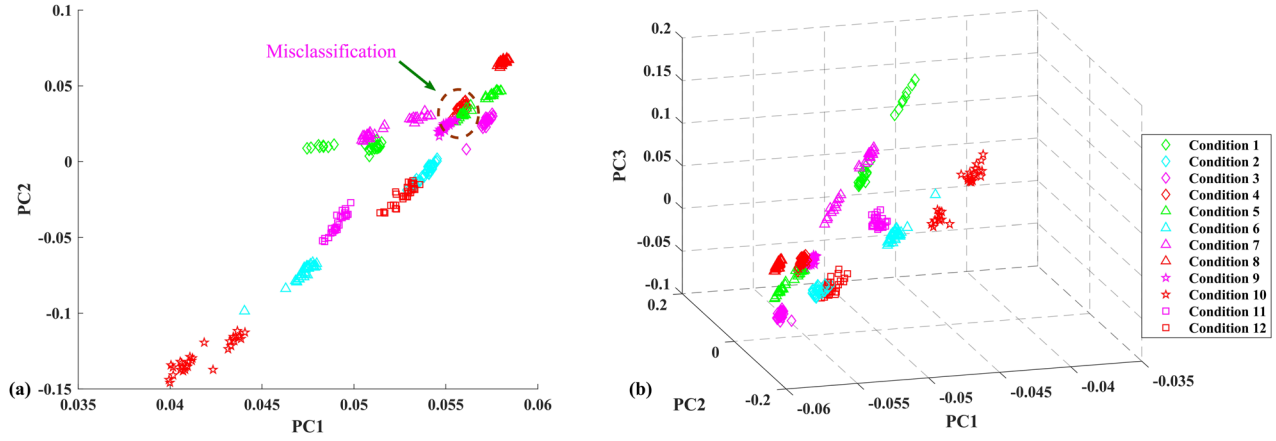

Figure 11. 2D-PCA and 3D-PCA projection of the second hidden-layer feature in experiment 1.

0.5, and the threshold is 0.01. The number of iterations of the first two RBMs is 300, and the number of iterations of the last RBM is 200.

The main purpose of experiment 1 is to classify multiple single-class faults of bearings by the proposed method and test their classification performance. The classification accuracy results for the 12 types of bearing conditions described in section 4.1 are shown in figures 7 and 8.

It can be seen from the above that the overall classification accuracy of SSA-DBN is 94.44%(340/360), the classification accuracy of the SADBN is 90.3%(325/360), and the classification accuracy of the DBN is 84%(303/360). The optimized SSA-DBN is better than the other two methods. Its optimal classification structure is 20-284-150-47-12. Figures 9–12 show the visual feature maps of experiment 1. PC1, PC2 and PC3 are the principal components of PCA, and each of

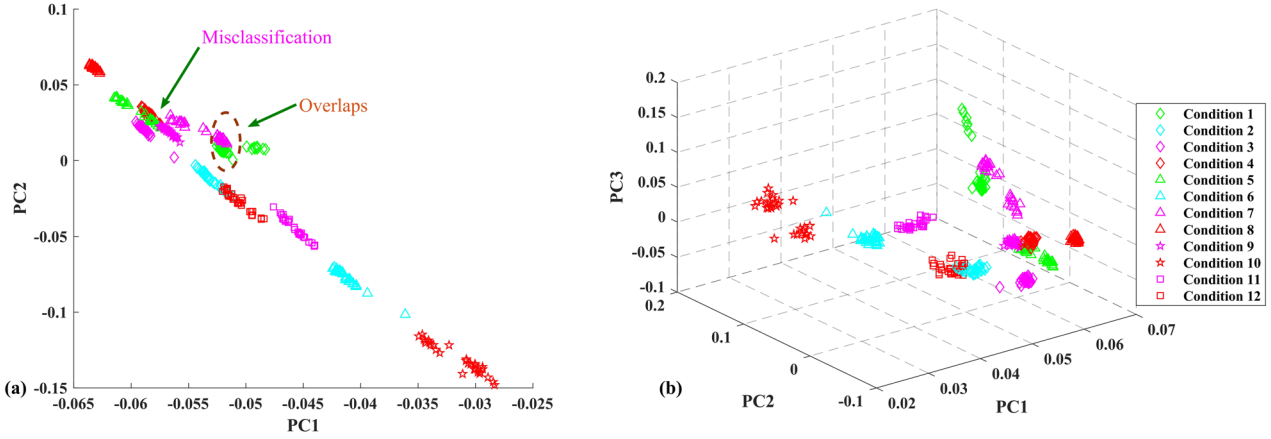

Figure 12. 2D-PCA and 3D-PCA projection of the third hidden-layer feature in experiment 1.

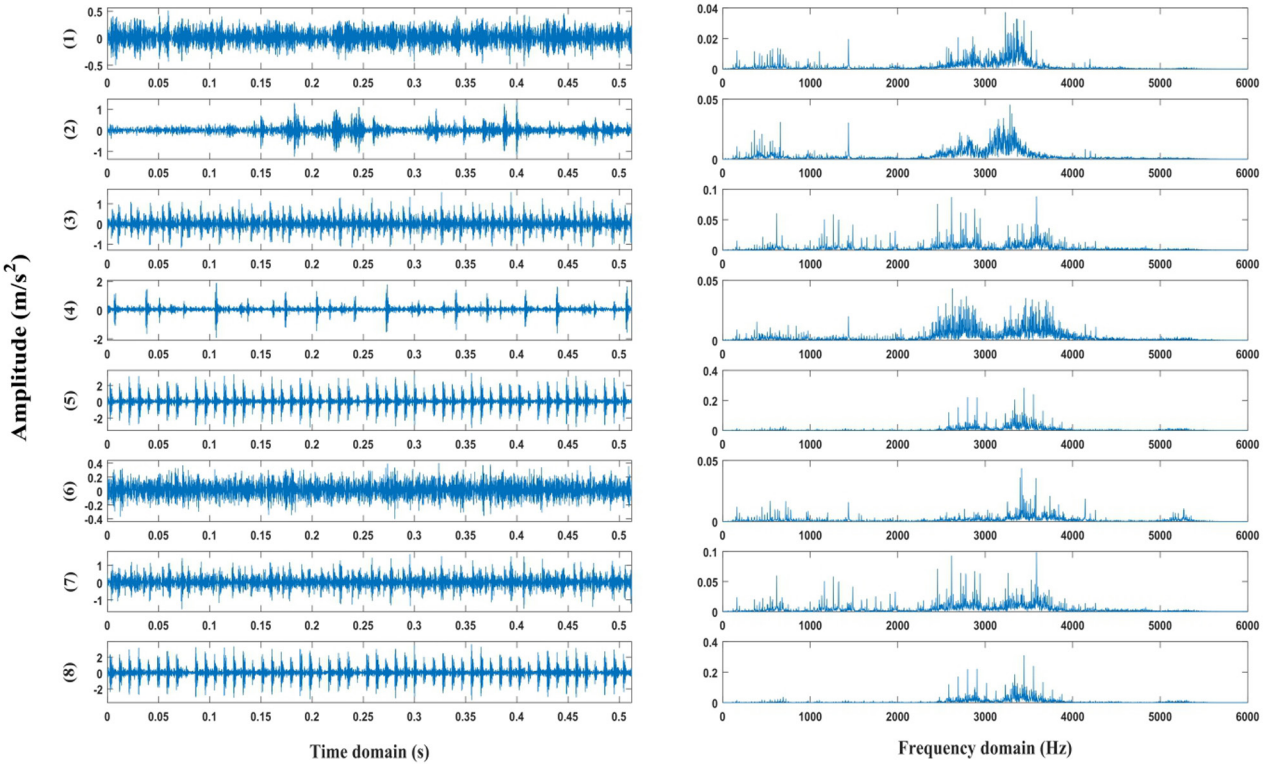

Figure 13. Time domain and corresponding frequency domain signal simulation of the eight kinds of bearing conditions.

them represents some information contained in the extraction feature.

The objective of experiment 1 is to classify 12-class bearing fault conditions. It can be seen from figures 9(a) and (b) that the extraction effect of DBN on the original features is relatively good. The advantage of the deep learning neural network is mainly reflected in the process of extracting raw features from vibration signals. Then, the layer-by-layer feature learning and classification of the DBN are performed. Figures 10(a) and (b) and 11(a) and (b) show that the 12-class samples are gradually separated on the basis of figure 9; however, there are still a few misclassifications in figures 10 and 11, which means that the classification is not effective. Figures 12(a) and (b) show a few misclassifications and overlaps, which means that the high-level features identify the real operating conditions of the bearing more easily than the original-level features.

4.4. Rolling bearing experiment 2

In experiment 2, eight kinds of bearing fault states are extracted: six single-point faults and two composite faults. The specific content is as follows: (1) Roller fault (0.007 in), (2) Roller fault (0.014 in), (3) Inner race fault (0.007 in), (4) Inner race fault (0.014 in), (5) Outer race fault (0.007 in, Center@6:00), (6) Outer race fault (0.014 in, Center@6:00), (7) Inner race (0.007 in) + Roller (0.007 in) composite failure, (8) Outer race (0.007 in, Center@6:00) + Roller (0.007 in) composite failure. The data used in this experiment are from Dataset2, and simulation signals for the time domain and

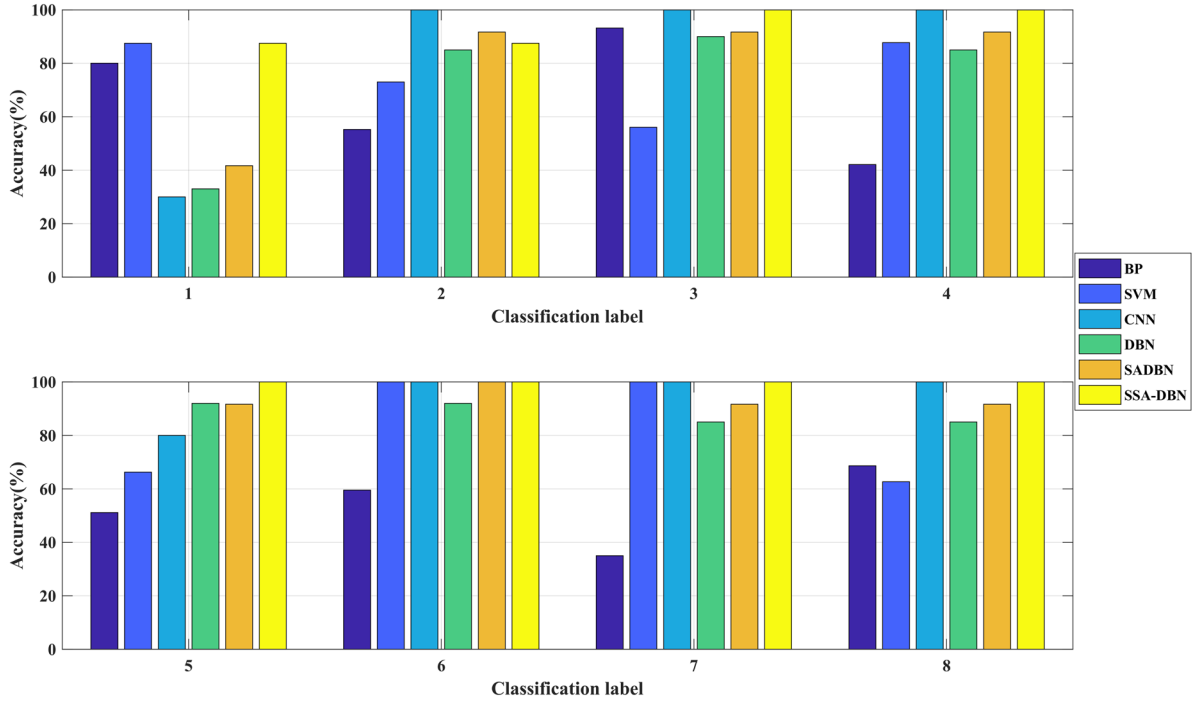

Figure 14. Comparative classification results of deep learning and shallow learning.

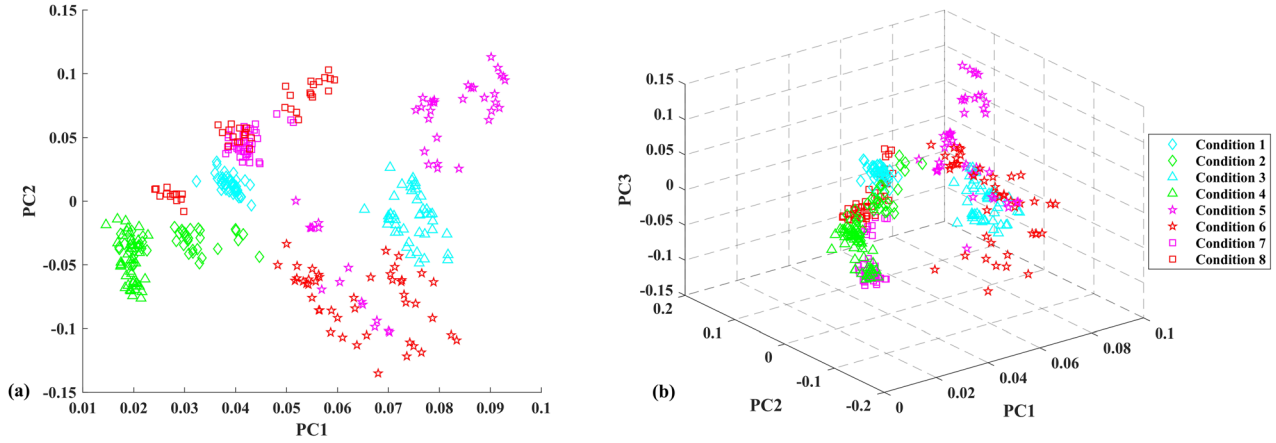

Figure 15. 2D-PCA and 3D-PCA projection of the original features in experiment 2.

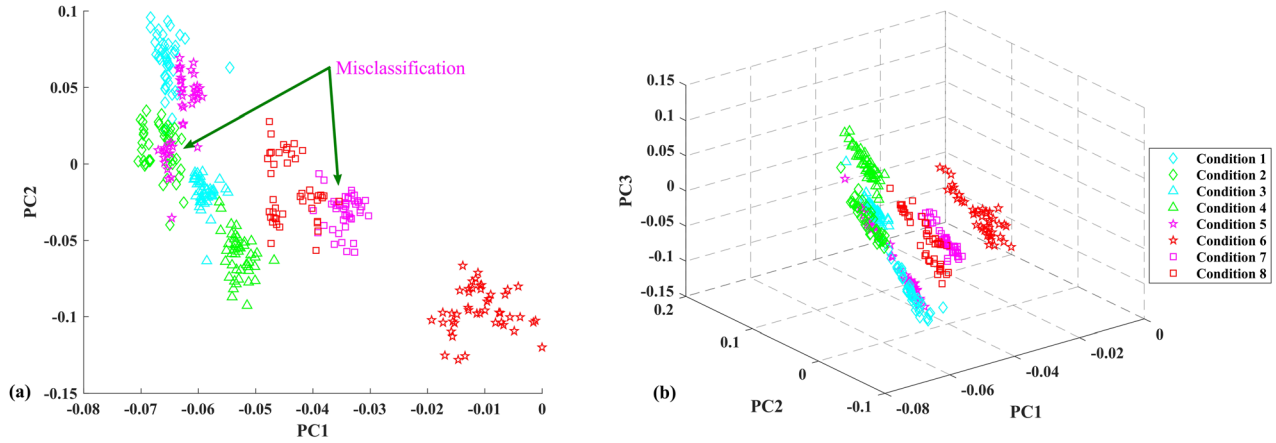

Figure 16. 2D-PCA and 3D-PCA projection of the first hidden-layer feature in experiment 2.

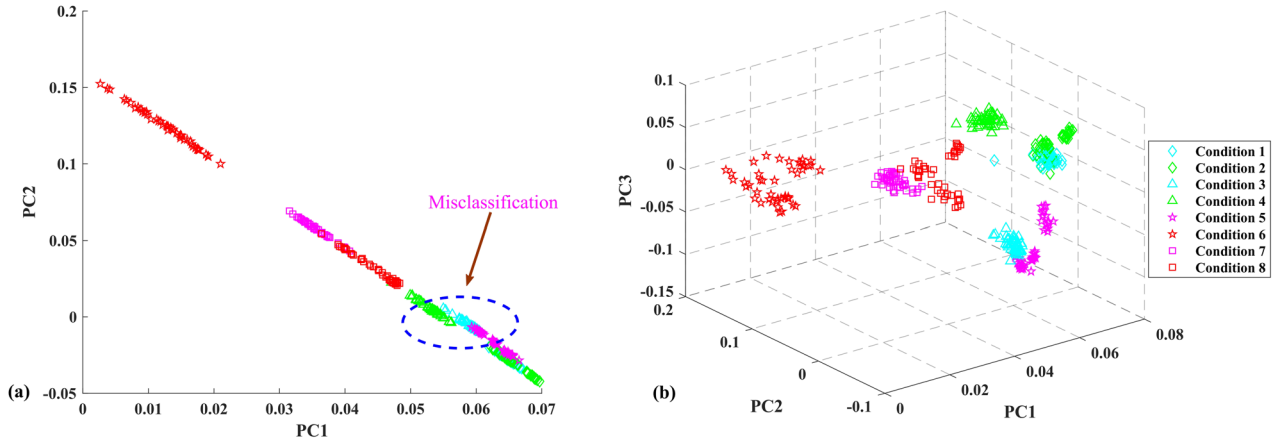

Figure 17. 2D-PCA and 3D-PCA projection of the second hidden-layer feature in experiment 2.

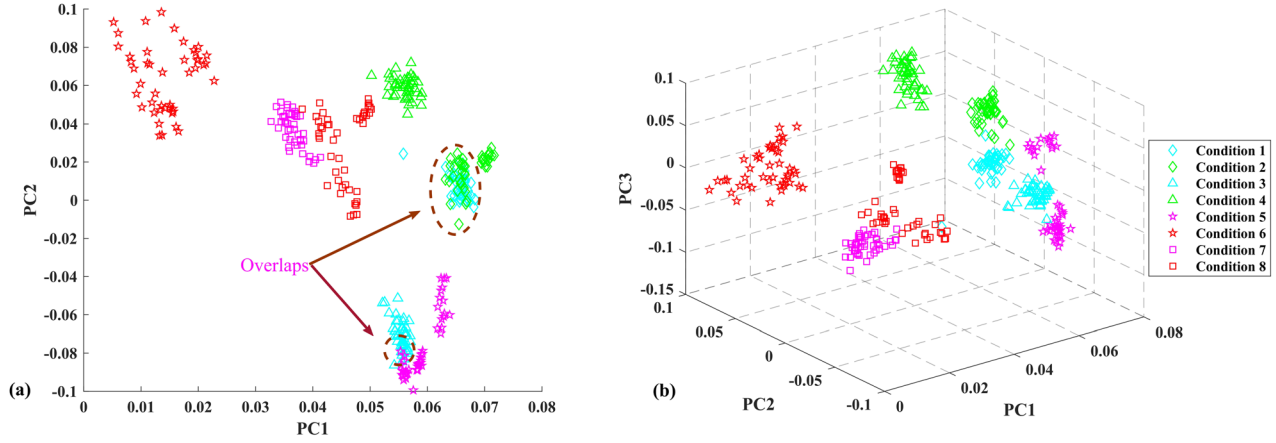

Figure 18. 2D-PCA and 3D-PCA projection of the third hidden-layer feature in experiment 2.

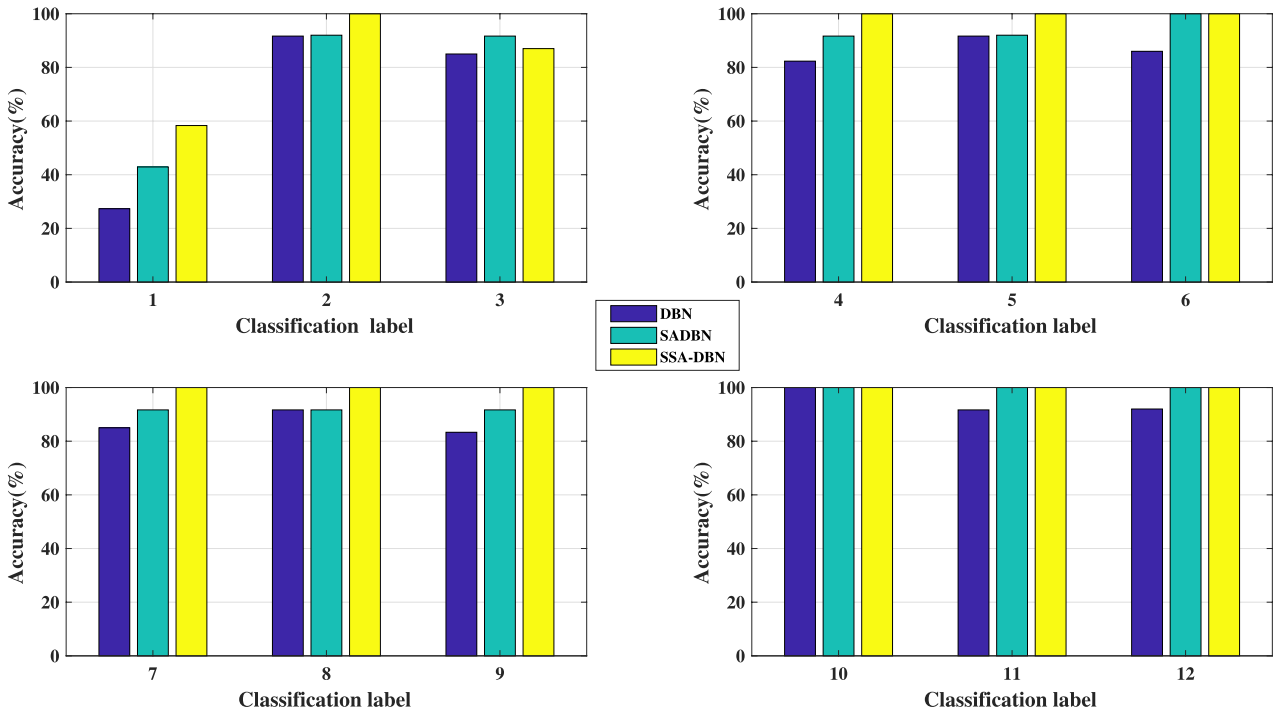

Figure 19. Fault classification results of experiment 3.

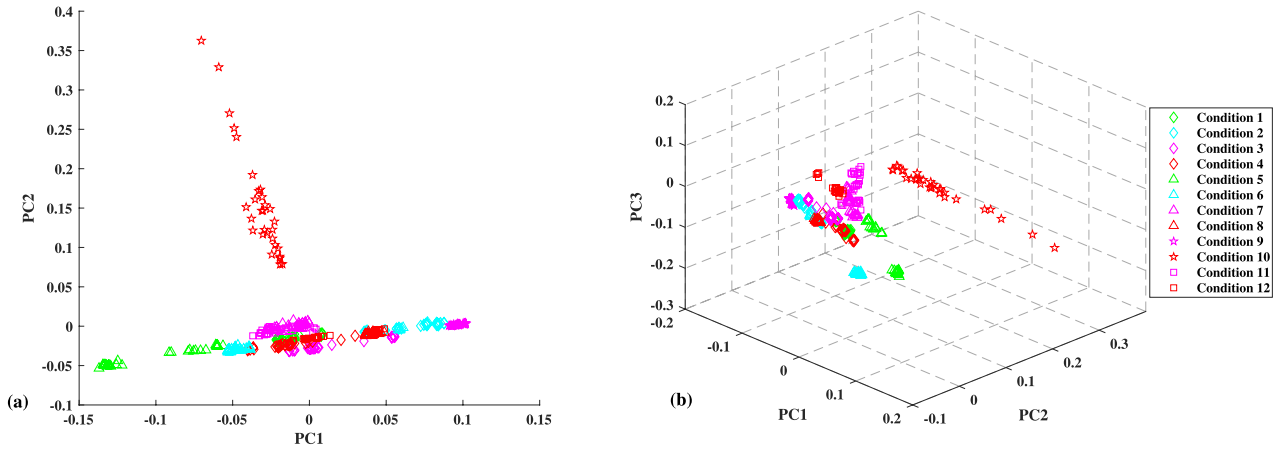

Figure 20. 2D-PCA and 3D-PCA projection of original features in experiment 3.

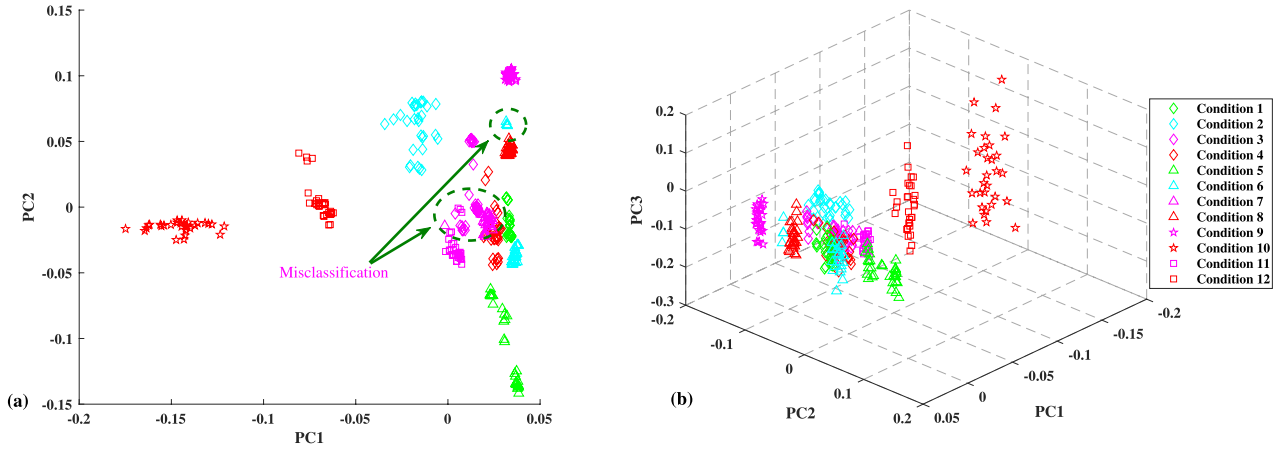

Figure 21. 2D-PCA and 3D-PCA projection of the first hidden-layer feature in experiment 3.

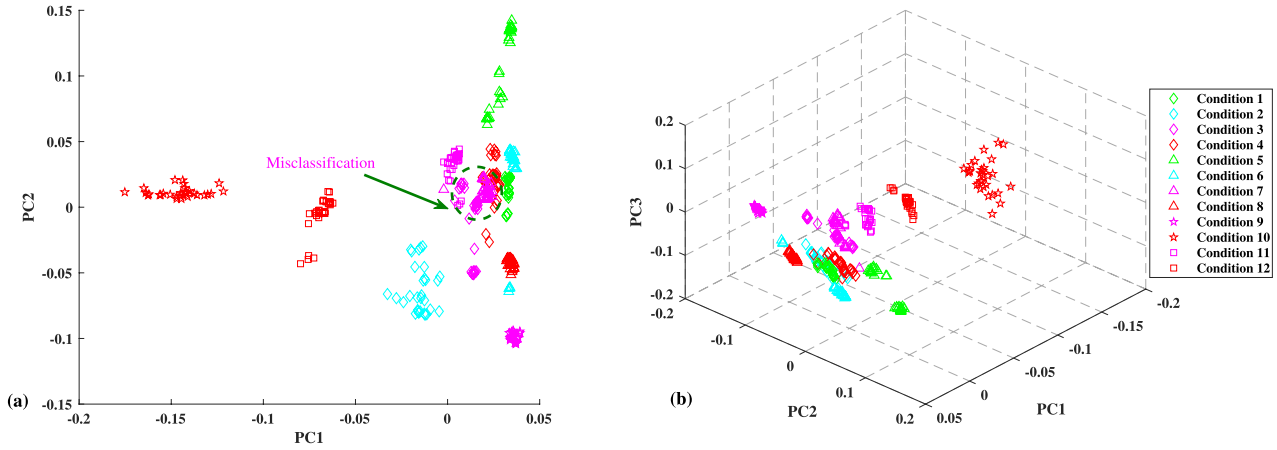

Figure 22. 2D-PCA and 3D-PCA projection of the second hidden-layer feature in experiment 3.

frequency domain corresponding to the eight bearing conditions are shown in figure 13. The experimental conditions for extracting data in this experiment are consistent with experiment 1. The main purpose of experiment 2 is to compare the superiority of the classification performance between the deep learning and the shallow learning cases.

Then, 25 feature parameters are extracted from the original feature set: nine time domain features, five frequency domain features, and 11 time-frequency domain features. The specific parameters of experiment 2 are as follows:

(1) **WPT:** The wavelet basis function is Meyer, and the number of decomposition layers is 3.
(2) **EMD:** EMD energy features of the first five IMF components.
(3) **LMD:** LMD energy spectrum features of the first five PF components.

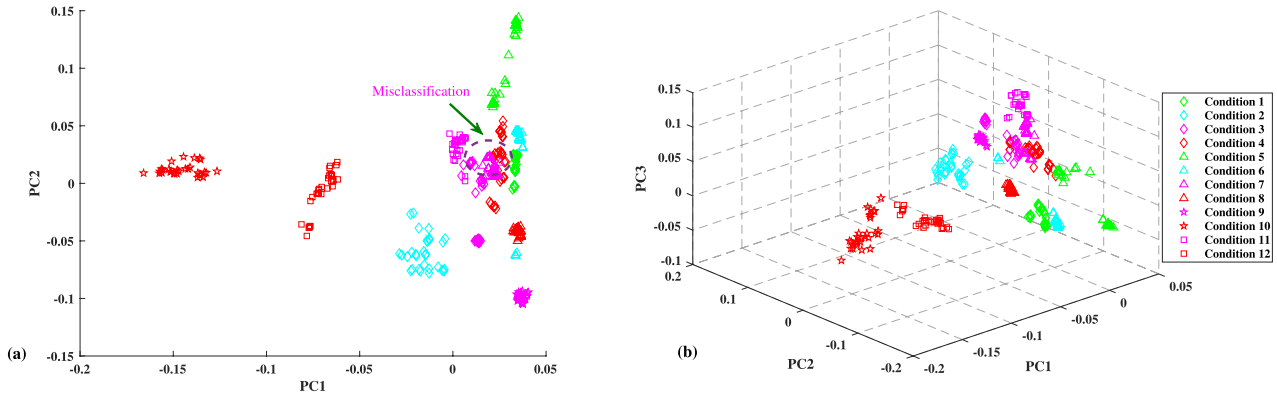

Figure 23. 2D-PCA and 3D-PCA projection of the third hidden-layer feature in experiment 3.

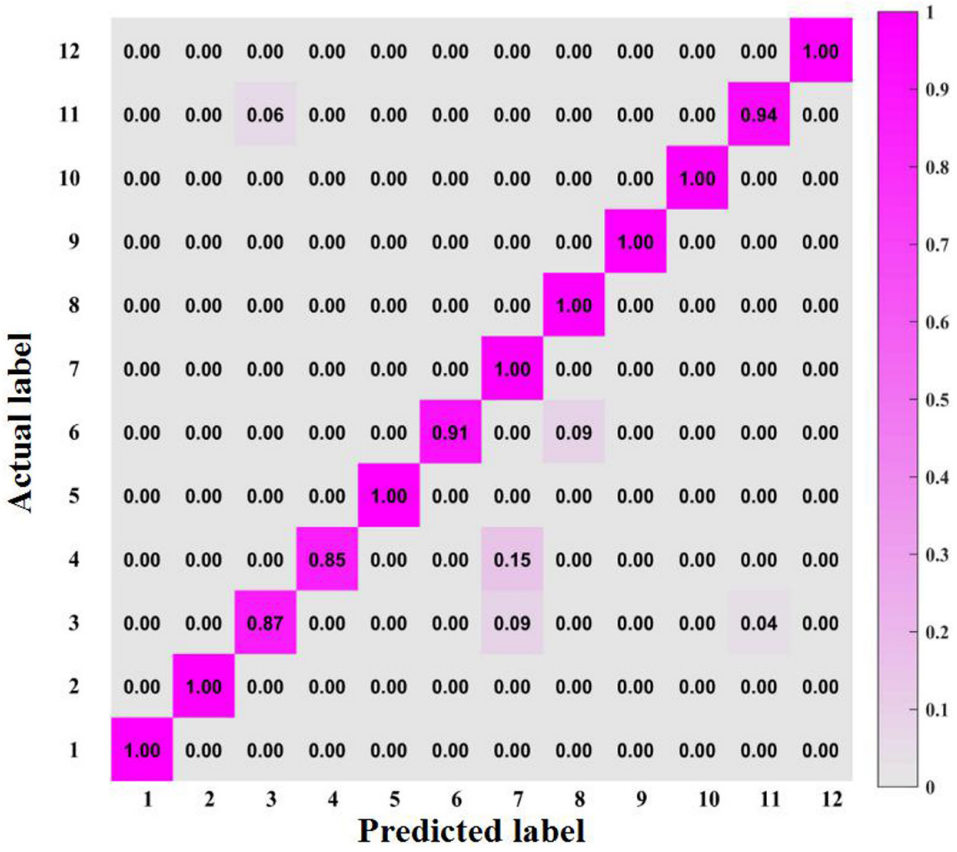

Figure 24. Multi-class hybrid matrix description.

(4) **SADBN:** The initial parameters of SADBN are the same as those of experiment 1.
(5) **CNN:** It consists of the Input layer, Convolution layer 1, Pooling layer 1, Convolution layer 2, Pooling layer 2 and Output layer [28]. The size of the input feature map is 24∗24, C1 layer contains six kernels, and C2 contains three kernels. The learning rate is 0.3, the step size is 2, and the number of iterations is 120.
(6) **BP:** The network structure is 25-45-8, the learning rate is 0.1, and the number of iterations is 500.
(7) **SVM:** The multi-layer perceptron kernel function is selected as a kernel function. The penalty factor is set to 25. The radius of the kernel function is 0.15.

Figure 14 shows the classification accuracy of the six single-point faults and the two composite faults of the bearing, and also demonstrates that the classification accuracy of BP and SVM is 63.61% and 82.53%, respectively. The classification accuracy of DBN, SADBN and CNN is 84%, 88.75% and 88.75%, respectively. The classification accuracy of SSA-DBN is 96.88%. Compared with other methods, the classification effectiveness of SSA-DBN is better, and the optimal structure is 25-140-70-36-8.

Experiment 2 presents the classification effect of eight bearing conditions. Figures 15(a) and (b) show the classification of the original features. Figures 16(a) and (b), and 17(a) and (b) show that as the number of hidden layers increases,

the misclassification gradually disappears. Figure 18(a) shows that there are still a few overlaps in the high-level hidden layer; however, figure 18(b) shows that the eight-class bearing conditions have been clearly classified, which means better classification. This overlap phenomenon demonstrates that the extracted feature information (principal component) is similar in some calculations.

4.5. Rolling bearing experiment 3

In experiment 3, 12 kinds of bearing fault states are extracted, and in addition the bearing state category is consistent with experiment 1. The selected bearing experimental data are from Dataset1 in table 6. Therefore, the main purpose of experiment 3 is to verify the effectiveness of this method-improved DBN (SSA-DBN) for the classification of bearing fault diagnosis under different load conditions. Moreover, the extracted feature parameters of this experiment are identical to those in experiment 1.

The fault classification results of experiment 3 are shown in figure 19. Therefore, the classification accuracy of the DBN is 86.8% (48/360), that of the SADBN is 91.1% (33/360), and that of the SSA-DBN is 96.2% (16/360).

Because the DBN itself has strong representation ability [29], PCA visualization technology is used to verify the classification of the SSA-DBN under different load conditions. Figures 20(a) and (b) show the 2D and 3D PCA visualization effects of the original characteristic parameters of the experimental data. Figures 21–23 show the output classification results of the first, second and third hidden layers of the SSA-DBN, respectively. As the number of hidden layers increases, the conditions of misclassification gradually decrease; nevertheless, a small number of misclassified parts mean that the classification effect is bad. As a whole, the overall classification effect is still rather good. In this experiment, the optimal network structure and Dataset1 are used to extract useful fault feature information, and the optimal structure of the SSA-DBN is 20-297-130-57-12. Figure 24 shows a probability matrix model for multiple categories of experiment 3. The abscissa is the predicted label and the ordinate is the actual label.

4.6. Experimental results and analysis

According to the above three experimental results, the following can be observed.

Experiment 1 first verifies the convergence problem of the SARBM proposed in this paper, and then applies the constructed SARBM to the DBN. It mainly verifies that the adaptive strategy proposed in this paper is superior to the performance of the standard RBM, and that the optimized SADBN has higher classification accuracy when dealing with multiple classes of single-fault problems.

Experiment 2 demonstrates that shallow learning methods (BP, SVM) mainly rely on artificial selection technology to complete the feature extraction, which greatly decreases the fault diagnosis effect. The method proposed in this paper not only solves the problem of artificially extracting features in shallow learning, but also improves the classification effectiveness in dealing with multiple classes of single and composite faults.

Experiment 3 mainly verifies the effectiveness of the SSA-DBN under different load conditions, and has a contrast with the results in Experiment 1.

5. Conclusion

Aiming at overcoming the difficulty of extracting useful information of characteristic parameters and the complexity of the rolling bearing structure and its running environment, a novel optimized SADBN (SSA-DBN) is proposed in this paper. This method pre-trains each SARBM by using the minimum batch SGD, and the BP neural network is used to inversely adjust its connection weight and bias. The SSA is used to optimize the network structure of the DBN. Meanwhile, the relationship between the number of neurons in each hidden layer summarized by many experiments can be used to determine the optimal structure of the neural network more quickly. The method proposed in this paper is applied to the three experiments, and the results show that its diagnostic effect is superior to that of other methods.

Acknowledgment

This research is supported by the National Natural Science Foundation of China (Grant No. U1708254) and the Liaoning Natural Science Fund Project (Grant No. 20170540725).

ORCID iDs

Yimin Zhang https://orcid.org/0000-0002-3526-0377

References

[1] McFadden P D and Smith J D 1984 Model for the vibration produced by a single point defect in a rolling element bearing *J. Sound Vib.* **96** 69–82
[2] Mcfadden P D and Smith J D 1985 The vibration produced by multiple point defects in a rolling element bearing *J. Sound Vib.* **98** 263–73
[3] Osman S and Wang W 2013 An enhanced Hilbert–Huang transform technique for bearing condition monitoring *Meas. Sci. Technol.* **24** 085004
[4] Yang J, Zhang Y and Zhu Y 2007 Intelligent fault diagnosis of rolling element bearing based on SVMs and fractal dimension *Mech. Syst. Signal Process.* **21** 2012–24
[5] Samanta B and Nataraj C 2009 Use of particle swarm optimization for machinery fault detection *Eng. Appl. Artif. Intell.* **22** 308–16
[6] Lin Y Y and Wang B W 2012 Application of a optimized wavelet neural networks in rolling bearing fault diagnosis *Appl. Mech. Mater.* **190** 919–22
[7] Zhu K, Song X and Xue D 2014 A roller bearing fault diagnosis method based on hierarchical entropy and support vector machine with particle swarm optimization algorithm *Measurement* **47** 669–75

[8] Rajeswari C, Sathiyabhama B, Devendiran S and Manivannan K 2014 Bearing fault diagnosis using wavelet packet transform, hybrid PSO and support vector machine *Procedia Eng.* **97** 1772–83
[9] Chen F, Tang B, Song T and Li L 2014 Multi-fault diagnosis study on roller bearing based on multi-kernel support vector machine with chaotic particle swarm optimization *Measurement* **47** 576–90
[10] Zhao Z Z, Xu Q S and Jia M P 2016 Improved shuffled frog leaping algorithm-based BP neural network and its application in bearing early fault diagnosis *Neural Comput. Appl.* **27** 375–85
[11] Liang M, Su D, Hu D and Ge M 2018 A novel faults diagnosis method for rolling element bearings based on ELCD and Extreme Learning Machine *Shock Vib.* **2018** 1891453
[12] Fu W, Tan J, Zhang X, Chen T and Wang K 2019 Blind parameter identification of MAR model and mutation hybrid GWO-SCA optimized SVM for fault diagnosis of rotating machinery *Complexity* **2019** 3264969
[13] Wei D, Jiang H, Shao H, Li X and Lin Y 2019 An optimal variational mode decomposition for rolling bearing fault feature extraction *Meas. Sci. Technol.* **30** 055004
[14] Ma P, Zhang H, Fan W, Wang C, Wen G and Zhang X 2019 A novel bearing fault diagnosis method based on 2D image representation and transfer learning-convolutional neural network *Meas. Sci. Technol.* **30** 055402
[15] Lei Y G, Jia F and Lin J 2016 An intelligent fault diagnosis method using unsupervised feature learning towards mechanical big data *IEEE Trans. Ind. Electron.* **63** 3137–47
[16] Schmidhuber J 2015 Deep learning in neural networks: an overview *Neural Netw.* **61** 85–117
[17] Hinton G E and Osindero S 2006 A fast learning algorithm for deep belief nets *Neurocomputing* **18** 1527–54
[18] Hinton G E and Salakhutdinov R R 2006 Reducing the dimensionality of data with neural networks *Science* **313** 504–7
[19] Huang W H, Song G J and Hong H K 2014 Deep architecture for traffic flow prediction: deep belief networks with multitask learning *IEEE Trans. Intell. Trans. Syst.* **15** 2191–201
[20] Shao H D, Jiang H K, Zhang X and Niu M G 2015 Rolling bearing fault diagnosis using an optimization deep belief network *Meas. Sci. Technol.* **26** 115002
[21] Shao H, Jiang H, Wang F and Wang Y N 2017 Rolling bearing fault diagnosis using adaptive deep belief network with dual-tree complex wavelet packet *ISA Trans.* 187–201
[22] Shao H, Jiang H, Zhang H J, Duan W J, Liang T C and Wu S P 2018 Rolling bearing fault feature learning using improved convolutional deep belief network with compressed sensing *Mech. Syst. Signal Process.* **100** 743–65
[23] Zhao X and Jia M 2018 A novel deep fuzzy clustering neural network model and its application in rolling bearing fault recognition *Meas. Sci. Technol.* **29** 125005
[24] Yamashita T, Tanaka M, Yoshida E, Yamauchi Y and Fujiyoshi H 2014 To be Bernoulli or to be Gaussian, for a restricted Boltzmann machine *Int. Conf. Pattern Recognit.* **22** 1520–5
[25] Mirjalili S, Gandomi A H and Mirjalili S Z 2017 Salp swarm algorithm: a bio-inspired optimizer for engineering design problems *Adv. Eng. Softw.* **114** 163–91
[26] Hayashida T, Nishizaki I, Sekizaki S, Nishida M and Prasetio M D 2017 Structural optimization of deep belief network by evolutionary computation methods including Tabu search *Trans. Mach. Artif.* **6** 69–80
[27] Erhan D, Bengio Y, Courville A C, Manzagol P A and Bengio S 2010 Why does unsupervised pre-training help deep learning? *J. Mach. Learn. Res.* **11** 625–60
[28] Lu C, Wang Z and Zhou B 2017 Intelligent fault diagnosis of rolling bearing using hierarchical convolutional network based health state classification *Adv. Eng. Inf.* **32** 139–51
[29] Bengio Y and Courville A 2013 Representation learning: a review and new perspectives *IEEE Trans. Softw. Eng.* **35** 1798–828