

Research Article

Design and Application of Deep Belief Network Based on Stochastic Adaptive Particle Swarm Optimization

Jianjian Yang , Boshen Chang, Xiaolin Wang, Qiang Zhang, Chao Wang, Fan Wang, and Miao Wu

School of Mechatronics and Information Engineering, China University of Mining and Technology, Beijing 100083, China

Correspondence should be addressed to Jianjian Yang; yangjiannedved@163.com

Received 24 March 2020; Revised 13 June 2020; Accepted 24 July 2020; Published 24 August 2020

Academic Editor: Kai Zhang

Copyright © 2020 Jianjian Yang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Due to the problem of poor recognition of data with deep fault attribute in the case of traditional superficial network under semisupervised and weak labeling, a deep belief network (DBN) was proposed for deep fault detection. Due to the problems of deep belief network (DBN) network structure and training parameter selection, a stochastic adaptive particle swarm optimization (RSAPSO) algorithm was proposed in this study to optimize the DBN. A stochastic criterion was proposed in this method to make the particles jump out of the original position search with a certain probability and reduce the probability of falling into the local optimum. The RSAPSO-DBN method used sample data to train the DBN and used the final diagnostic error rate to construct the fitness value function of the particle swarm algorithm. By comparing the minimum fitness value of each particle to determine the advantages and disadvantages of the model, the corresponding minimum fitness value was selected. Using the number of network nodes, learning rate, and momentum parameters, the optimal DBN classifier was generated for fault diagnosis. Finally, the validity of the method was verified by bearing data from Case Western Reserve University in the United States and data collected in the laboratory. Comparing BP (BP neural network), support vector machine, and heterogeneous particle swarm optimization DBN methods, the proposed method demonstrated the highest recognition rates of 87.75% and 93.75%. This proves that the proposed method possesses universality in fault diagnosis and provides new ideas for data identification with different fault depth attributes.

1. Introduction

Machine learning is a popular research interest in artificial intelligence and pattern recognition. Its theory and methods have been widely applied to complex problems in engineering applications and science [1–4]. There have also been important achievements in the fault diagnosis of mechanical equipment as many domestic and foreign scholars have conducted in-depth researches and achieved fruitful results in the field. Some machine learning methods used are shown in (Figure 1). For example, the main intelligent diagnostic methods include support vector machine (SVM), artificial neural network (ANN), multilayer perceptron (MLP), kernel method (KMs), and other pattern recognition methods [5–8]. These methods have achieved desirable results in the fault diagnosis of mechanical equipment, but they belong to the algorithm structure called “shallow learning.” Here, the

function fitting must be completed in one or two layers of the model’s structure; thus, fault diagnosis results are unstable [9, 10]. Moreover, a long latency period of the equipment does not imply sudden failure because there is a long critical state before the failure occurs. The depth of component wear or scratches has not reached the state where serious failure occurred for a while. Sample data of various fault depth attributes have been extracted, but the traditional shallow model has been ineffective in characterizing complex nonlinear mapping relationships between signals and devices. The recognition effect of the shallow network is poor in the case of semisupervised and weakly marked. Hence, developing new fault depth detection methods is imperative.

Deep learning is an emerging machine learning method, which mainly simulates the structure of the human brain and achieves efficient data processing through hierarchical learning.

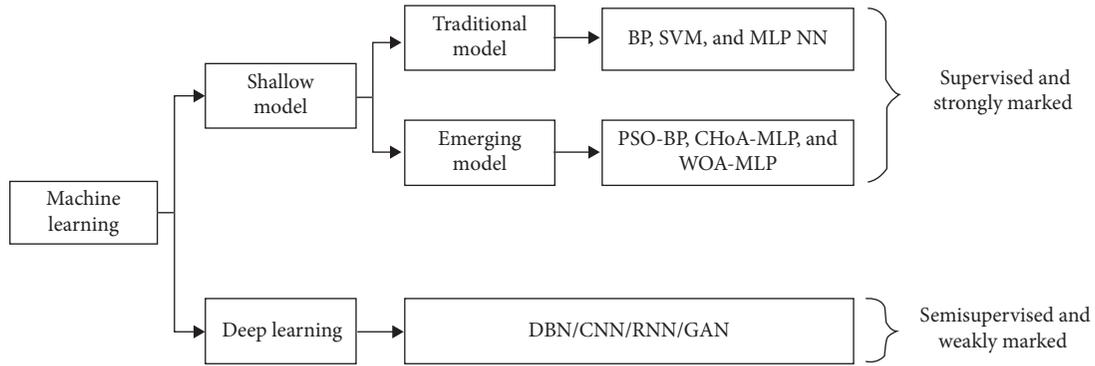


FIGURE 1: Machine learning methods.

The deep belief network proposed by Hinton et al. is a classic algorithm in deep learning [11], which opens the door to deep learning. For instance, Tamilselvan et al. [12] applied it to monitor and identify the health status of machinery and equipment, and experiments proved that it can effectively identify the fault status of equipment. Compared with other deep learning models such as CNN, RNN, and GAN, DBN can more easily capture data features under big data and has the advantage of good compatibility with other algorithms [13]. Compared with the shallow model, DBN can use the initial stacked restricted Boltzmann machine to unsupervise feature extraction and then use the classifier to fine-tune [14], which can improve the classification effect in the case of semisupervised and weak labeling. Further, Sun et al. [15] used signal processing to extract the fault characteristics of the monitoring signal and used deep learning to diagnose the type of mechanical failure and the degree of damage. Shan Waiping [16] also studied the reconstruction and feature extraction of the original vibration signals of rolling bearings by the DBN. Shao et al. [17] proposed a DBN for time-domain feature extraction and particle swarm optimization (PSO) for the fault diagnosis of rolling bearings. However, these studies have only set DBN's structural parameters based on experience or repeated experiments. The model optimization training takes time and is difficult to achieve accurate fault diagnoses.

Therefore, to improve the accuracy of fault diagnosis and reduce the optimization time of the model, this study proposes a fault diagnosis method based on the stochastic adaptive particle swarm algorithm (RSAPSO) and DBN. Particle swarm optimization is a swarm intelligence global optimization search algorithm, which has been well applied in neural network parameter optimization [18]. Compared with other intelligent optimization algorithms, PSO has fewer parameters, is easier to implement, and can be more accurate results [19]. Using the parallel search capability of the RSAPSO, the model parameters of DBN were optimized and selected. This method used DBN training on the sample data to construct a fitness function. The final recognition error rate was used as the termination condition of the improved particle swarm algorithm iteration. The RSAPSO is associated with the parameter optimization of the DBN to effectively generate a suitable classifier to improve the accuracy of fault diagnosis rate.

2. Stochastic Adaptive Particle Swarm Optimization Deep Belief Network

2.1. DBN Training Process. The DBN is a stack of multiple RBMs. Its structure is shown in Figure 2. The lower layer represents the details of the original data, and the upper layer represents the data attribute category or feature. The data is abstracted layer by layer from the lower layer upward. The DBN can deeply dig the essential characteristics of the data. It reduces the impact of human factors and effectively improves the training results of neural networks.

The core pretraining method of DBN is the greedy layer-by-layer learning algorithm, which trains each layer of RBM separately and is unsupervised. That is, RBM1 is completely trained before RBM2 is trained. RBM is unsupervised training and has no expected output. Its role is to extract features and adjust the training parameters of the model. Its node value is binary: 0 or 1.

After completing the RBM pretraining, the entire DBN preliminary structure is formed, but a labeled adjustment link to the DBN is still required. A BP classifier is set at the last layer of the DBN network. The fine-tuning process is supervised, and the weights and biases obtained by the RBM pretraining are adjusted to make the network training more accurate. This improves the recognition accuracy of the network.

The steps of the DBN fault diagnosis method are outlined in Figure 3.

Initially enter all samples, including marked and unmarked. Extract features by stacking RBM, and then use labeled sample labels for fine-tuning to achieve semi-supervised fault diagnosis.

2.2. Stochastic Adaptive Particle Swarm Algorithm. In the standard PSO algorithm, the weight adjustment formula has larger limitations, and the adjustment range of the weight w is smaller. Therefore, the shortcomings of local optimization and low search accuracy often occur. To improve the search accuracy of the algorithm and reduce the probability of falling into a local optimum, this study adopts an RSAPSO algorithm, which introduces stochastic adaptation in the algorithm. This allows particles to stochastically reset the position with a certain probability, thus jumping out of the

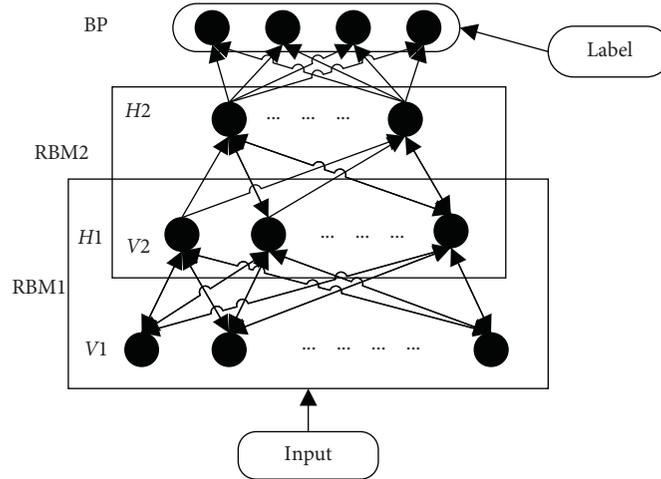


FIGURE 2: DBN structure.

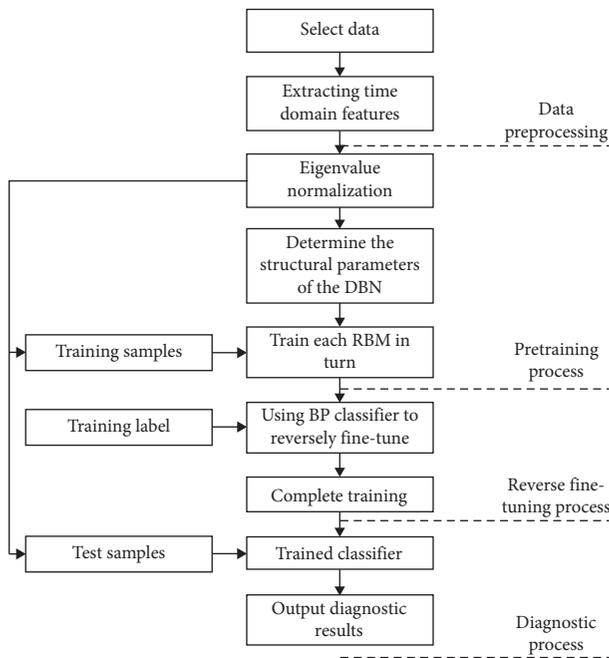


FIGURE 3: DBN troubleshooting process.

original position and search again. Consequently, the probability that the particle swarm is trapped in a local minimum is reduced.

To begin with, the RSAPSO algorithm modifies the inertia weight formula of the PSO algorithm, as shown in

$$w = w_{\max} - \frac{(w_{\max} - w_{\min}) \times (f_{\max} - \text{fitness}(p))}{f_{\max} - f_{\min}}, \quad (1)$$

where w_{\max} and w_{\min} are the maximum and minimum values of inertia weight, respectively; $\text{fitness}(p)$ is the fitness value of the p th particle of each generation; f_{\max} and f_{\min} are the maximum and minimum fitness values of each generation of particles, respectively.

This method can automatically adjust the parameters of the inertia weight w according to the current particle fitness

value. When the fitness value is large, w becomes large, which can increase the particle search speed and improve the global search ability of the particle. Conversely, when the fitness value is small, w becomes small, which can reduce the particle search speed and improve the local search ability of the particle. The modified inertial weight w has a larger adjustment range and improves the searchability of the algorithm, as shown in Figure 4.

Second, the particle swarm algorithm does not have a process such as cross-mutation, and it tends to fall into the local optimum; therefore, the stochastic rule is added. When the stochastic number in the update formula is greater than the set threshold, let the particles stochastically reset the position with a certain probability, thereby reducing the probability of the particle swarm falling into the local minimum. The stochastic criterion is shown in

$$x_{id} = \begin{cases} x_{\max} \times \text{rands}(1, D), & \text{rands} \leq P, \\ x_{id}, & \text{rands} \geq P, \end{cases} \quad (2)$$

where P is the threshold, x_{\max} is the maximum allowed position, x_{id} is the current position, and $\text{rands}(1, D)$ is the D stochastic number between 0 and 1.

2.3. RSAPSO-DBN Model. A neural network has a tendency to train new samples and forget old samples during training, and too many types of training will cause a low learning efficiency and slow convergence speed. Therefore, it is necessary to choose a suitable momentum parameter m and learning rate η . When the values of m and η are too large, the corresponding update weights W and thresholds a and b will increase, which will increase the convergence speed. However, it will make the model unstable, the loss function will continually oscillate, and it is difficult to improve the accuracy, as shown in Figure 5. As shown at 1 in Figure 5, when the values of m and η are too small, W , a , and b will become smaller, which will cause the model to converge slowly and require longer training time, and the values may fall into the local maximum during reverse fine-tuning. This results in model training failure, as shown at 2 in Figure 5. Therefore,

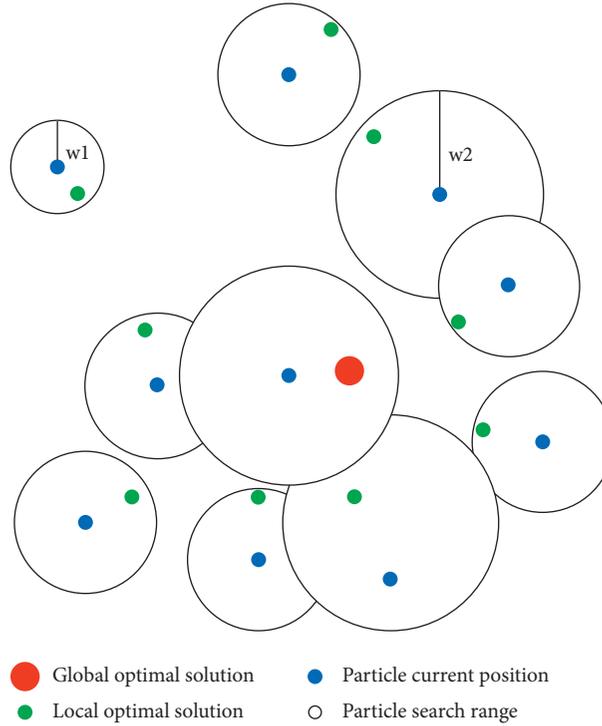


FIGURE 4: Schematic of optimal solution search.

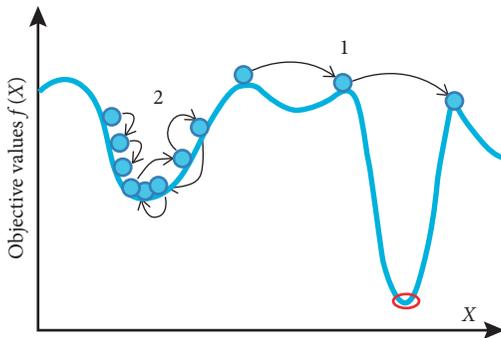


FIGURE 5: Optimization process.

the need to choose the optimal momentum parameters and learning rate is the key to a successful model training.

After the training set and classification are determined, the corresponding input and output layer nodes are determined accordingly. A large number of experiments show that if the number of nodes in the hidden layer N is too small, the network cannot have the necessary learning and information processing capabilities. Conversely, if it is too large, it will not only greatly increase the complexity of the network structure, but the network will be more likely to fall into a local minimum during the learning process, which will make the learning speed of the network very slow.

As there are currently no methods for determining the nodes of a deep neural network, the learning rate η , the momentum parameter m , and the number of nodes N in the hidden layer, most studies still determine the network

structure parameters of the DBN based on experience or multiple experiments. The optimization parameters of particle swarm are learning rate η , momentum m , and number of hidden layer nodes N .

After the population is initialized, the parameter analysis space is set, and then iteration starts. When the number of iteration steps is less than the maximum number of steps, the fitness value of each particle will be continually calculated and the minimum fitness value among them will be recorded and checked against the convergence claim. When the convergence requirement is reached, the loop is skipped and the training is completed, but when the requirement is not met, iteration is continued until the maximum number of iteration steps is reached. Finally, the parameters of the minimum fitness value are recorded, and the DBN classifier is output.

The optimization process is shown in Figure 6.

3. Network Optimization Analysis

As stated already, the particle swarm optimization neural network has three parameters: learning rate η , momentum parameter m , and number of hidden layer nodes N . For the network optimization analysis, the number of particles is set. Generally, 20–40 particles are selected, and the particle dimension is defined as 3, corresponding to the three parameters. That is, each particle is represented as $p(m_p, \eta_p, N_p)$.

The RBM model is an energy-based model that defines an energy function that is used to introduce a series of probability distribution functions:

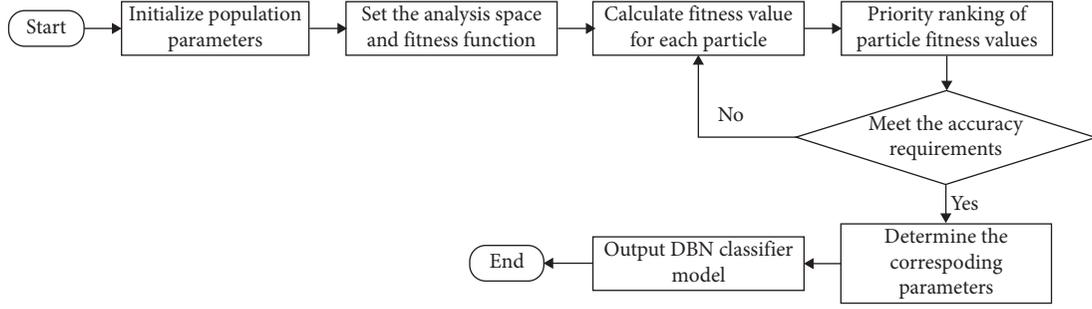


FIGURE 6: Population optimization flowchart.

$$E_{\theta}(v, h) = - \sum_{i=1}^{n_v} a_i v_i - \sum_{j=1}^{n_h} b_j h_j - \sum_{i=1}^{n_v} \sum_{j=1}^{n_h} v_i w_{ij} h_j, \quad (3)$$

where n_v represents the number of nodes in the visible layer; n_h represents the number of nodes in the hidden layer, which represents the weight value of the i th layer of the visible layer to the j th node of the hidden layer; $\theta = \{W, a, b\}$ represents the set of all parameters of the system.

Using the above energy function formula, when θ is determined, the joint probability of (v, h) can be obtained according to the energy function as

$$P_{\theta}(v, h) = \frac{e^{-E_{\theta}(v, h)}}{z_{\theta}}, \quad (4)$$

$$z_{\theta} = \sum_{v, h} e^{-E_{\theta}(v, h)},$$

where z_{θ} is a normalized term that guarantees that P becomes a probability distribution and a partition function.

The conditional probability of the hidden unit i given the visible unit vector v and the conditional probability of the visible unit j given the hidden unit vector h are

$$P = (h_j = 1 | v) = \sigma \left(b_j + \sum_{i=1}^{n_v} w_{ji} v_i \right), \quad (5)$$

$$P = (v_i = 1 | h) = \sigma \left(a_i + \sum_{j=1}^{n_h} w_{ij} h_j \right).$$

From the calculated probability distribution, Gibbs sampling is used to extract a sample $h_1 \sim P(h_1 | v_1)$; h_1 is used to reconstruct the visible layer (i.e., the hidden layer) to infer the visible layer. Then, the apparent layer's activation probability $P(v_i = 1 | h_1)$ is calculated. Similarly, a Gibbs sampling is taken from the calculated probability distribution to extract a sample $v_2 \sim P(v_2 | h_2)$. Then, the activation probability $P(h_j = 1 | v_2)$ of each neuron in the hidden layer by v_2 is calculated. This was performed for k Gibbs samplings. The weights and thresholds are updated according to

$$\begin{cases} W(t+1) = m_p W(t) + \eta_p (v_i^0 h_j^0 - v_i^k h_j^k), \\ a(t+1) = m_p a(t) + \eta_p (v_i^0 - v_i^k), \\ b(t+1) = m_p b(t) + \eta_p (h_j^0 - h_j^k). \end{cases} \quad (6)$$

The parameters $p(m_p, \eta_p, N_p)$ of 40 particles are initialized, the analytic space of m_p, η_p, N_p is defined, and the fitness function of the particle swarm algorithm is defined as the DBN recognition error rate. Further, the parameter values of 40 particles are put into the DBN for testing, the fitness value corresponding to each particle is calculated, the particle p with the smallest fitness value is selected, and its corresponding $p_1(m_p, \eta_p, N_p)$ is copied. This is performed iteratively for 40 particles: the fitness value corresponding to each particle is calculated, and the particle parameter $p_2(m_p, \eta_p, N_p)$ corresponding to the minimum fitness value is selected. This continues until the particle fitness value reaches the requirement or the maximum number of iterations, $p(m_p, \eta_p, N_p)$ corresponding to the last minimum fitness value is selected, the parameter value is assigned to the DBN network, and the optimal DBN is output.

Using a particle swarm algorithm to select network parameters not only avoids the uncertainty of parameter value selection caused by human experience but also improves the implementation efficiency of the network.

4. Sample Data Testing and Analysis

4.1. Bearing Data Experiment of Case Western Reserve University

4.1.1. Data Description. The bearing data of Case Western Reserve University in the United States is available to the public, and bearing failure data of the fan end was adopted in this study. The sampling frequency was 12 kHz. The normal load and the outer ring failure (OF-6 o'clock direction) under the condition of 0 loads (speed 1797r/min) were selected as (0.007, 0.014, and 0.021 inches), inner ring failure (IF-0.007, 0.014, and 0.021 inches), rolling body failure (RF-0.007, 0.014, and 0.021 inches), and 4 types and 3 types of fault depth. The rotation speed and sampling frequency were calculated to obtain the vibration acceleration of 401 points collected per revolution. More than two points were used for calculations in this study. To facilitate the calculation, 1000 sampling points were used as a group for time-domain analysis to extract valid values after 17 types of time-domain features, including variance, peak value, kurtosis index, and margin index. Then, 200 sets of samples were obtained for each type of fault data. Among these, 160 were selected as training samples, and the remaining 40 were tested samples, as shown in Figure 7.

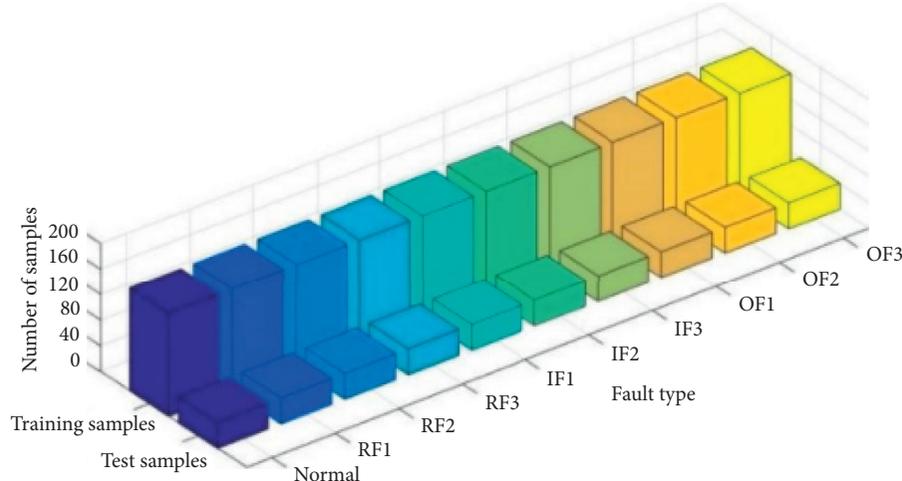


FIGURE 7: Data characterization.

A total of 1600 training samples and 400 test samples were obtained for this study.

4.1.2. Experimental Verification. The data in Section 3.1.1 was used to input BP, SVM, standard PSO, optimized APSO, and the improved RSAPSO-optimized DBN network used in this study for 10 tests. Among these, the DBN's hidden layer node analysis space was [10,20], the learning rate parsing space was (0,0.1], and the momentum parameter parsing space was [0.8,1). The recognition rates of the five different models are shown in Figure 8.

It can be seen from Figure 8 that the shallow network's BP and SVM struggled to identify data with different fault depth attributes. The recognition rate is only 50% once, and the others are below 50%. Both PSO-DBN and APSO-DBN only exceeded the RSAPSO-DBN recognition rate once, and the highest recognition rate was lower than the RSAPSO-DBN proposed in this study. It can be seen that the actual effect of the proposed method is due to the other two PSO methods. More precisely, the structural parameters of the DBN were defined. The parameters of the optimal result particle swarm optimization DBN are shown in Table 1.

4.2. Experimental Analysis of Different Fault Depth Databases Based on Support Vector Data Description (SVDD). In reality, it is difficult to obtain the data of different fault depths of the same fault type. Hence, it is a serious limitation of this kind of research. Thus, the authors proposed a linear sample data generation method based on SVDD for generating data with different fault depth attributes.

SVDD is a description method based on boundary data. The goal is to find a minimal sphere or domain that contains all or almost all target samples. When it is difficult to obtain different degrees of damage to the same part, SVDD is used to describe the fault data. Thus, abnormal data with different degrees of damage are constructed for further fault diagnosis research.

The linear weight selection was selected according to the time-domain characteristic formula. Taking the peak as an

example, the envelope contrast images of two different fault depths were drawn, as shown in Figure 9.

As shown in Figure 9, when the fault depth is 0.007 inches, its vibration is relatively stable, and at a fault depth of 0.021 inches, its peak value relatively increases by a certain linear ratio. Therefore, for a single sample of data, different depth attribute data are generated by increasing a certain linear ratio. This is based on the relationship between the recognition accuracy and the linear sample library generated here, and the fault state is shown in equation (11). The supersphere description image is shown in Figure 10. One has

$$Z(A) = \begin{cases} \text{minor failure} & A \geq 70\%, \\ \text{intermediate failure} & 30 < A < 70\%, \\ \text{serious failure} & A \leq 30\%. \end{cases} \quad (7)$$

where A is the recognition accuracy during construction.

Use a single sample dataset IF1/OF1/RF1 to create a trained SVDD model. The single sample dataset is increased by a certain linear ratio and input to SVDD for judgment, and the linear weight is judged according to equation (7), thereby constructing a linear sample database.

4.2.1. Construction of Deep Sample Library for Laboratory Data. This study collected the bearing data in the laboratory rotating machine test bench. The experimental platform is shown in Figures 11 and 12. The bearing data is divided into 4 types: normal, rolling element failure (RF), inner ring failure (IF), and outer ring failure (OF). The data sampling frequency is 5.12 kHz. Figure 11 shows the red circle marks for sensor sampling.

The authors selected three levels of data: 100%, 50%, and 0% for illustration. The relationship between recognition accuracy and linear weight is shown in Figures 13–15. The specific values are shown in Table 2.

4.2.2. Experimental Verification. The linear data sample library constructed in Section 3.2.2 was used for shallow

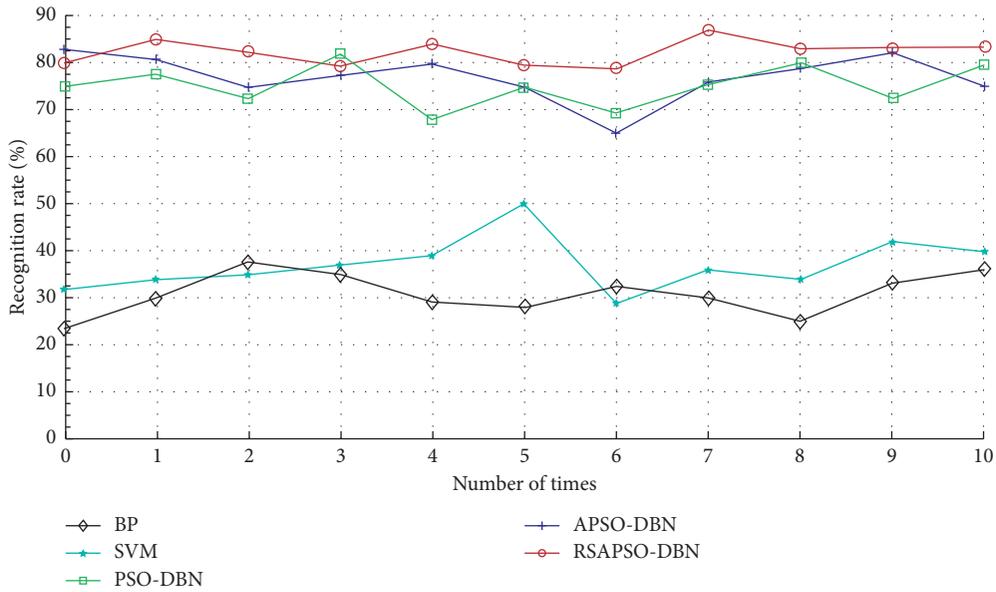


FIGURE 8: Recognition rate of each method in 10 experiments.

TABLE 1: Heterogeneous particle swarm DBN diagnosis results.

Method	PSO-DBN	APSO-DBN	RSAPSO-DBN
Number of hidden layer nodes	15	15	15
Learning rate	0.0902	0.0842	0.095
Momentum parameter	0.8965	0.9356	0.8462
Recognition accuracy	81.75%	83%	87%

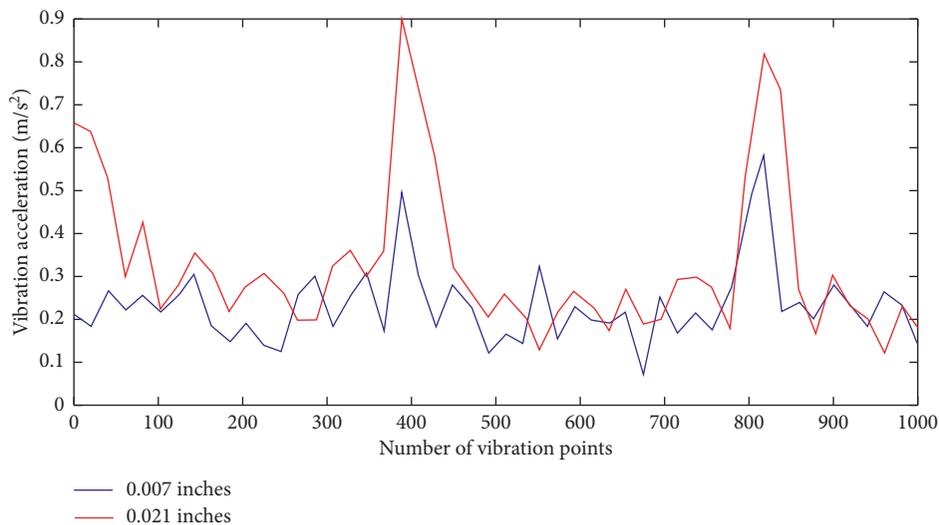


FIGURE 9: Comparative image of faults at different depths.

networks and three PSO-optimized DBN diagnostics. The DBN’s hidden layer node parsing space was [10, 20] and the learning rate parsing space was (0, 0.1]. The parsing space of the momentum parameter was [0.8,1). The line chart of the recognition rate after 10 tests is shown in Figure 16. The structural parameters of the DBN corresponding to the optimal result are shown in Table 3. Each type of data was

selected, making 160 sets of training samples and 40 sets of test samples. The test results are shown in Table 3. The line chart of the highest recognition rate of the three particle swarm optimization algorithms is shown in Figure 17.

According to the test results in Table 3 and Figure 15, it can be seen that, in the generated linear sample database with different degrees of damage, DBN can well identify

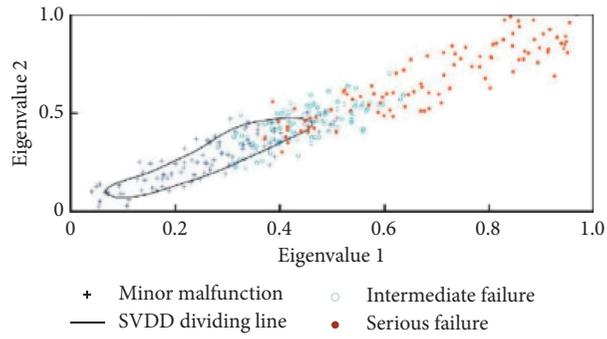


FIGURE 10: SVDD fault type image.



FIGURE 11: Test platform controls.



FIGURE 12: Overall view of the test platform.

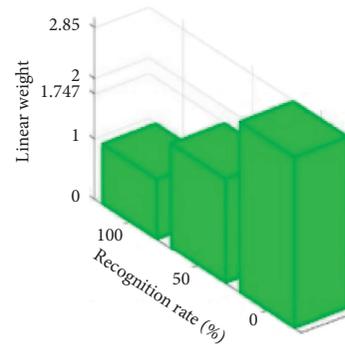


FIGURE 14: Linear structure of IF.

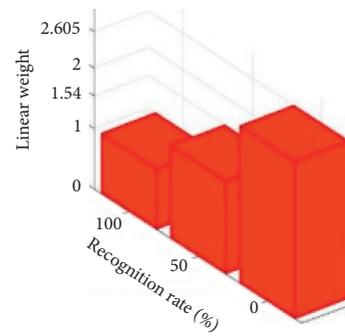


FIGURE 15: Linear structure of OF.

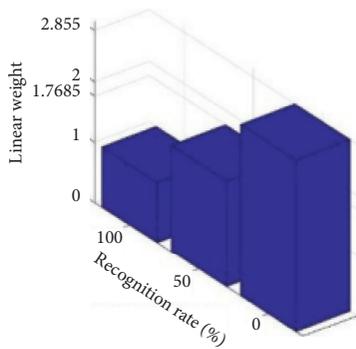


FIGURE 13: Linear structure of RF.

TABLE 2: Linear sample database.

Fault type	Recognition accuracy (%)	Linear weight
Normal	100	1
RF1	100	1
RF2	50	1.7685
RF3	0	2.855
IF1	100	1
IF2	50	1.747
IF3	0	2.85
OF1	100	1
OF2	50	1.54
OF3	0	2.605

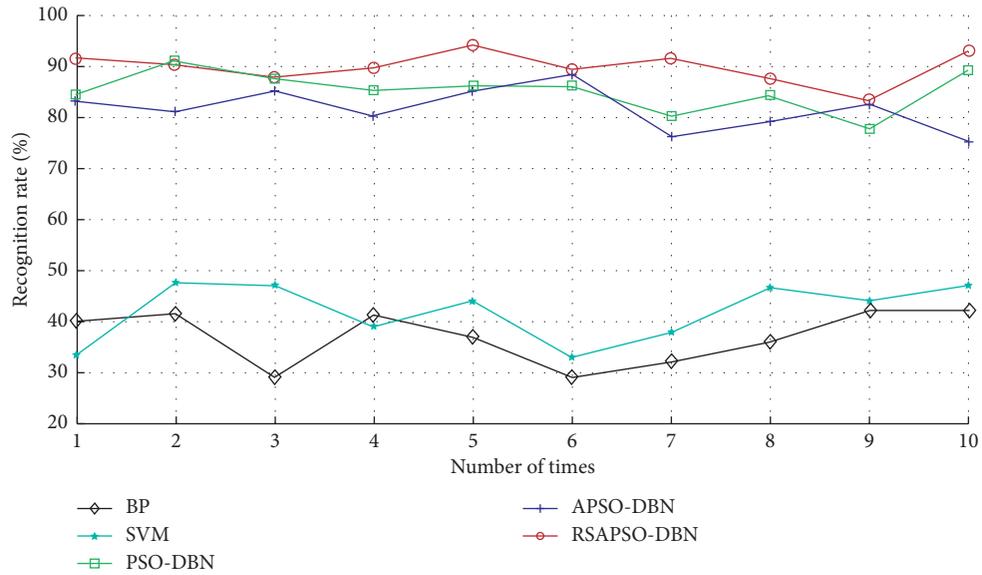


FIGURE 16: Laboratory data test accuracy.

TABLE 3: Structural parameters of heterogeneous particle swarm DBN.

Method	PSO-DBN	APSO-DBN	RSAPSO-DBN
Number of hidden layer nodes	15	18	16
Learning rate	0.0536	0.0236	0.0167
Momentum parameter	0.943	0.8931	0.8762
Recognition accuracy	90a.75%	88.25%	93.75%

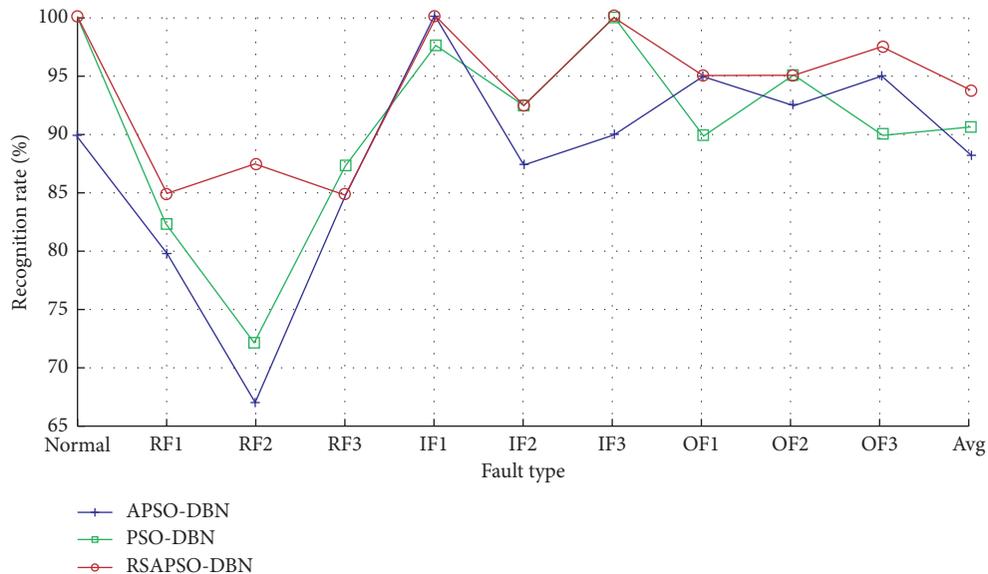


FIGURE 17: Line chart of heterogeneous PSO-DBN recognition accuracy.

various types of data, and BP and SVM can hardly identify data with different fault depth attributes. It can also be proven that DBN has a superior recognition effect of different fault depth data. From Figure 16, it can be seen that,

among the 10 types of fault type and average recognition rates, RSAPSO-DBN has the best effect. Only one fault recognition rate is lower than PSO-DBN; hence, it is proven that the RSAPSO optimization method proposed in this

paper is superior in more cases. The DBN structure parameter defined thereby has the best performance and can better identify data with different fault depth attributes.

5. Conclusions

It is difficult for a shallow network to identify data with different fault depth attributes. Hence, a deep neural network is required for fault diagnosis. Compared to a shallow network, the deep network does not fall into a local optimal situation, which can effectively perform fault depth identification. Due to the inaccuracy caused by a large number of experiments or empirically defined methods for the DBN network structure parameters, the improved particle swarm algorithm proposed in this study was compared with traditional algorithms. Results show that its parallel search ability can effectively determine the network structure parameters of DBN compared to custom parameters as it improves efficiency and accuracy.

Meanwhile, the proposed method has been used in most mechanical equipment such as bearings, motors, and roadheaders in modern machinery. By analyzing the rules of vibration signals presented by mechanical equipment, faults can be diagnosed and the state can be detected. There are still some problems with the method proposed in this paper, which increases the complexity of the network model, resulting in an increase in its single run time and an increase in the number of calculation steps, which also puts a certain burden on the equipment; the population optimization algorithm is ultimately external. The optimization algorithm needs to further define the structural parameters from the principle of deep learning; at present, this method is only used in fault diagnosis of mechanical equipment, and it needs to be further extended to other fields of machine learning. Therefore, how to solve the above problems will be the next key research direction of DBN research.

Data Availability

Experimental data were collected on a rotating machine test bench inside the China University of Mining and Technology (Beijing) Laboratory. Data cannot be shared publicly due to some confidentiality reasons.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This paper was supported by the National Natural Science Foundation of China (Grant nos. 2018101030061 and 2018101060080) and Shanxi Special Fund for Science and Technology (Grant no. 20181102027).

References

- [1] M. Khishe and M. R. Mosavi, "Improved whale trainer for sonar datasets classification using neural network," *Applied Acoustics*, vol. 154, pp. 176–192, 2019.
- [2] M. Khishe and M. R. Mosavi, "Classification of underwater acoustical dataset using neural network trained by Chimp Optimization Algorithm," *Applied Acoustics*, vol. 157, 2020.
- [3] X. Yang, L. Kong, and Z. Liu, "Machine learning and deep learning methods for cybersecurity," *IEEE Access*, vol. 99, p. 1, 2018.
- [4] G. Wang, J. C. Ye, and K. Mueller, "Image reconstruction is a new frontier of machine learning," *IEEE Transactions on Medical Imaging*, vol. 99, p. 1, 2018.
- [5] M. Song, H. Song, and S. Xiao, "A study on fault diagnosis method of rolling bearing based on wavelet packet and improved BP neural network," *IOP Conference Series: Materials Science and Engineering*, vol. 274, Article ID 012133, 2017.
- [6] J. Yang, Z. Tang, Z. Wang, and M. Wu, "Fault diagnosis on cutting unit of mine roadheader based on PSO-BP neural network," *Coal Science and Technology*, vol. 45, no. 10, pp. 129–134, 2017.
- [7] M. Khishe and M. R. Mosavi, "Improved whale trainer for sonar datasets classification using neural network," *Applied Acoustics*, vol. 154, pp. 176–192, 2019.
- [8] K. H. Hui, M. H. Lim, M. S. Leong, and S. M. Al-obaidi, "Dempster-Shafer evidence theory for multi-bearing faults diagnosis," *Engineering Applications of Artificial Intelligence*, vol. 57, pp. 160–170, 2017.
- [9] W. Chenglin and B. Zhejing, "A review of data driven-based incipient fault diagnosis," *Acta Automatica Sinica*, vol. 42, no. 9, pp. 1285–1299, 2016.
- [10] L. Yaguo, J. Feng, Z. Xin et al., "A deep learning-based method for machinery health monitoring with big data," *Journal of Mechanical Engineering*, vol. 51, no. 21, pp. 49–56, 2015.
- [11] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.
- [12] P. Tamilselvan, W. Yibin, and W. Pingfeng, "Deep belief network based state classification for structural health diagnosis," *Reliability Engineering and System Safety*, vol. 115, no. 3, pp. 124–135, 2013.
- [13] G. Zhong, H. Wang, K. Zhang, B. Jia et al., "Fault diagnosis of Marine diesel engine based on deep belief network," in *Proceedings of the 2019 Chinese Automation Congress (CAC)*, pp. 3415–3419, Hangzhou, China, 2019.
- [14] Y. Wei and Z. Weng, "Research on TE process fault diagnosis method based on DBN and dropout," *The Canadian Journal of Chemical Engineering*, vol. 98, no. 6, 2020.
- [15] J. Sun, R. Wyss, A. Steinecker et al., "Automated fault detection using deep belief networks for the quality inspection of electromotors," *tm-Technisches Messen*, vol. 81, no. 5, pp. 255–263, 2014.
- [16] S. Waiping and Z. Xueqiong, "Signal reconstruction and bearing fault identification based on deep belief network," *Electronic Design Engineering*, vol. 24, no. 4, pp. 67–71, 2016.
- [17] H. Shao, H. Jiang, X. Zhang et al., "Rolling bearing fault diagnosis using an optimization deep belief network," *Measurement Science & Technology*, vol. 26, no. 11, Article ID 115002, 2015.
- [18] Q. Qin, Y. Feng, and L. I. Feng, "Structural reliability analysis using enhanced cuckoo search algorithm and artificial neural network," *Journal of Systems Engineering and Electronics*, vol. 29, no. 6, pp. 1317–1326, 2018.
- [19] H. Quan, D. Srinivasan, and A. Khosravi, "Short-term load and wind power forecasting using neural network-based prediction intervals," *Ieee Transactions on Neural Networks & Learning Systems*, vol. 25, no. 2, pp. 303–315, 2017.