

Article

Bearing State Recognition Method Based on Transfer Learning Under Different Working Conditions

Ning Cao ¹, Zhinong Jiang ^{1,*}, Jinji Gao ² and Bo Cui ³

¹ National Defense Key Laboratory of Ministry of Education, Beijing University of Chemical Technology, Beijing 100029, China; 18811757532@163.com

² Diagnostic and Self-Healing Engineering Research Center, Beijing University of Chemical Technology, Beijing 100029, China; gaojinji@263.net

³ College of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100029, China; 13146696006@163.com

* Correspondence: jiangzhinong@263.net; Tel.: +86-139-1158-9618

Received: 3 December 2019; Accepted: 27 December 2019; Published: 31 December 2019



Abstract: Bearing state recognition, especially under variable working conditions, has the problems of low reusability of monitoring data, low state recognition accuracy and low generalization ability of the model. The feature-based transfer learning method can solve the above problems, but it needs to rely on signal processing knowledge and expert diagnosis experience to obtain the cross-characteristics of different working conditions data in advance. Therefore, this paper proposes an improved balanced distribution adaptation (BDA), named multi-core balanced distribution adaptation (MBDA). This method constructs a weighted mixed kernel function to map different working conditions data to a unified feature space. It does not need to obtain the cross-characteristics of different working conditions data in advance, which simplifies the data processing and meet end-to-end state recognition in practical applications. At the same time, MBDA adopts the *A-Distance* algorithm to estimate the balance factor of the distribution and the balance factor of the kernel function, which not only effectively reduces the distribution difference between different working conditions data, but also improves efficiency. Further, feature self-learning and rolling bearing state recognition are realized by the stacked autoencoder (SAE) neural network with classification function. The experimental results show that compared with other algorithms, the proposed method effectively improves the transfer learning performance and can accurately identify the bearing state under different working conditions.

Keywords: transfer learning; multi-core balanced distribution adaptation; SAE neural networks; rolling bearing; different working condition

1. Introduction

Modern industrial production technology makes great contributions to improving productivity, reducing losses, saving natural resources and human resources, reducing the scrap rate, and ensuring product quality. As a key piece of equipment, rotary machinery is widely applied in important engineering fields, such as power, electric power, chemical, metallurgy, mining and machinery manufacturing. Once large-scale mechanical equipment fails, it will cause huge economic losses and even cause different degrees of casualties. As a key component of rotating machinery, rolling bearings play an important role in ensuring safe and efficient operation of the machine. The working condition of the rolling bearing not only affects the operation of the machine itself, but also the subsequent production. According to statistics, in all rotating machinery faults, bearing failure accounts for about 30% [1]. Therefore, it is of great significance for the continuous production system to realize the state recognition of rolling bearings. The rapid development of signal analysis and processing technology,

computer technology and network technology, and artificial intelligence provides technical support for state recognition and fault diagnosis of rolling bearings.

Early bearing state detection and fault diagnosis methods have used the vibration mechanism of rolling bearings and signal analysis and processing techniques to extract features, and have then utilized expert diagnostic experience to achieve bearing fault diagnosis and state recognition [2–7]. These methods laid the foundation for the development of bearing diagnostics. However, these methods rely too much on feature extraction and expert experience. In recent years, machine learning and deep learning have achieved good results in the fields of image recognition and natural language process. More and more scholars have applied them to the state recognition of rolling bearings [8–11]. As a special machine learning, deep learning has the characteristics of nonlinearity, multi-layer and adaptability. Therefore, it has powerful feature extraction capabilities that can automatically extract features from original data. Fault diagnosis models of rolling bearings based on deep learning mainly include convolutional neural networks (CNN) [12,13], long-term short-term memory neural networks (LSTM) [14,15], deep belief networks (DBN) [16,17], and stacked autoencoders (SAE) [18]. Compared with other network models in state recognition, SAE networks have noise reduction filtering and feature extraction functions. In particular, the SAE network with the classification function can achieve higher state recognition accuracy in small samples, which fully reflects the powerful feature extraction function and robustness of the method. The above method achieves good results in the state recognition of the rolling bearing. In practical applications, the working environment of rolling bearings is very complicated, and the working conditions often change. Under variable working conditions, training samples and test samples do not meet the same distribution conditions. Therefore, the above method will no longer apply. How to accurately identify the bearing state under different working conditions is an urgent problem to be solved.

Since the NIPS Symposium “Learning to Learn” in 1995, transfer learning has attracted more and more attention in the field of deep learning [19]. As a cross-data, cross-model and cross-task learning method, transfer learning has achieved the classification of target data by applying knowledge learned from source data [20]. Yosinski et al. [21] first studied transfer learning of deep learning and pointed out that transfer learning can solve problems such as less target data and uneven distribution of data. The feature-based transfer learning method is one of the important methods for transfer learning, mainly including transfer component analysis (TCA) [22], joint distribution adaptation (JDA) [23] and balanced distribution adaptation (BDA) [24]. In [25], the variational mode decomposition (VMD) and semi-supervised transfer component analysis (SSTCA) were proposed to achieve a bearing state classification under variable conditions. In [26], the subspace alignment (SA) was proposed to minimize the distribution difference of the domain data. In [27], a least squares support vector machine (LSSVM) transfer learning strategy based on auxiliary data was used for bearing fault diagnosis under different working conditions. The above bearing state recognition methods based on transfer learning have achieved good results, but require feature transformation or feature selection of bearing data.

Inspired by the above research, this paper uses the BDA method to transfer the vibration characteristics of rolling bearings under variable conditions. However, BDA has the following two problems: (1) The balance factor μ needs to be obtained by searching in the experiment; and (2) it needs to obtain some cross-characteristics of different domain data in advance. In order to solve the above problems, this paper proposes the MBDA method by introducing mixed kernel functions, which can extract richer features of the data without feature transformation. At the same time, the balance factor of the kernel function γ is introduced, which directly affects the effect of mixed kernel functions on data mapping. This paper adopts the A – Distance algorithm to calculate the balance factor of the distribution μ and the balance factor of the kernel function γ , which effectively minimize the distribution divergence between different domain data. Then SAE neural network is used to realize the feature self-learning and state recognition of the rolling bearing.

This paper is organized as follows. Section 2 introduces the basic principle of the BDA algorithm and SAE neural network, and gives the definition of our new algorithm with the aim to ensure that it

is understandable. Section 3 gives a flow chart of the bearing diagnostic algorithm for the proposed method. We performed experiments to show the good performance of the proposed method in the bearing state identification in Section 4, and a conclusion is presented in Section 5.

2. Basic Theory of the Methodology

2.1. Balanced Distribution Adaptation

As a cross-domain, cross-model and cross-task learning method, transfer learning can effectively solve the problem of different distributions of source data and target data. Assuming the original space is X and the class label is Y , the labeled source data and the unlabeled target data are $D_S = \{X_S, Y_S\}$ and $D_T = \{X_T, Y_T\}$, respectively, where $X_S, X_T \in X$ and $Y_S, Y_T \in Y$. The marginal probability distributions and conditional probability distributions of source data and target data respectively are $P(X_S), P(Y_S/X_S)$ and $P(X_T), P(Y_T/X_T)$. In practical applications, the marginal probability distributions and conditional probability distributions of source data and target data are not equal, i.e., $P(X_S) \neq P(X_T)$ and $P(Y_S/X_S) \neq P(Y_T/X_T)$. The goal of transfer learning methods is to minimize the marginal and conditional distribution discrepancy between source data and target data. BDA as a transfer learning method can adaptively minimize the marginal and conditional distribution discrepancy between domains by exploiting a balance factor μ . The formula is as follows:

$$\text{Distance}(X_S, X_T) = (1 - \mu) \|P(X_S), P(X_T)\| + \mu \|P(Y_S/X_S), P(Y_T/X_T)\| \quad (1)$$

BDA uses the maximum mean difference (MMD) to calculate the distribution difference between the two domains. The method assumes that there is a mapping function ϕ , which satisfies $P(\phi(X_S)) \approx P(\phi(X_T))$ and $P(Y_S/\phi(X_S)) \approx P(Y_T/\phi(X_T))$. Therefore, Equation (1) can be represented as:

$$\text{Distance}(X_S, X_T) = (1 - \mu) \left\| \frac{1}{n_s} \sum_{i=1}^{n_s} \phi(x_{s_i}) - \frac{1}{n_t} \sum_{j=1}^{n_t} \phi(x_{t_j}) \right\|_H^2 + \mu \sum_{c=1}^C \left\| \frac{1}{n_s^{(c)}} \sum_{x_i \in D_S^{(c)}} \phi(x_i) - \frac{1}{n_t^{(c)}} \sum_{x_j \in D_T^{(c)}} \phi(x_j) \right\|_H^2 \quad (2)$$

where n_s and n_t are the number of samples of source data and target data, respectively, and C is the number of categories.

By further taking advantage of matrix tricks and regularization, Equation (2) can be represented as:

$$\begin{aligned} \min & \text{tr} \left(A^T X \left((1 - \mu) M_0 + \mu \sum_{c=1}^C M_c \right) X^T A \right) + \lambda \|A\|_F^2 \\ \text{S.T.} & \quad A^T X H X^T A = I, \quad 0 \leq \mu \leq 1 \end{aligned} \quad (3)$$

where the first term represents the marginal and conditional distribution divergences between the two domains, A denotes the transformation matrix, and λ is the regularization parameter. The constraints $A^T X H X^T A = I$ ensure that the transformed data $A^T X$ maintain the important properties of X_S and X_T . $I_{n_s+n_t} \in \mathbb{R}^{(n_s+n_t) \times (n_s+n_t)}$ is the identity matrix, and $H = I_{n_s+n_t} - (1/n_s + 1/n_t) \mathbf{1}\mathbf{1}^T$ is the central matrix. $\mu \in [0, 1]$ is estimated by searching in the experiment. The calculation formula for each element of the M_0 and M_c is as follows:

$$(M_0)_{ij} = \begin{cases} \frac{1}{n_s^2}, & x_i, x_j \in D_S \\ \frac{1}{n_t^2}, & x_i, x_j \in D_T \\ -\frac{1}{n_s n_t}, & \text{otherwise} \end{cases} \quad (4)$$

$$(M_c)_{ij} = \begin{cases} \frac{1}{n_s^{(c)} n_s^{(c)}}, & x_i, x_j \in D_S^{(c)} \\ \frac{1}{n_t^{(c)} n_t^{(c)}}, & x_i, x_j \in D_T^{(c)} \\ -\frac{1}{n_s^{(c)} n_t^{(c)}}, & \begin{cases} x_i \in D_S^{(c)}, x_j \in D_T^{(c)} \\ x_i \in D_T^{(c)}, x_j \in D_S^{(c)} \end{cases} \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

The above non-convex optimization problem is transformed into the trace optimization problem by the Lagrange multiplier method, and the specific process will not be described.

2.2. Multi-Core Balanced Distribution Adaptation

BDA has the following two problems: (1) it is inefficient to get the balance factor μ ; and (2) it needs to obtain some cross-characteristics of different domain data in advance. In order to solve the above problems, this paper proposes the MDBA method. The kernel function plays an important role in the BDA. The effect of a single kernel function in transfer learning is not ideal. Weighted mixed kernel functions combine the advantages of different kernel functions, but add a new parameter γ . The formula for weighted mixed kernel functions is as follows:

$$\begin{aligned} K'(x_i, x_j) &= \gamma K_{RBF}(x_i, x_j) + (1 - \gamma) K_{Ploy}(x_i, x_j) \\ &= \gamma \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) + (1 - \gamma) [(x_i \cdot x_j) + 1]^d \end{aligned} \quad (6)$$

where K_{RBF} and K_{Ploy} are the radial basis function (RBF) and polynomial kernel (Ploy) function respectively, and $\gamma \in [0, 1]$ controls the weight of the two kernel function. The formula of K_{RBF} and K_{Ploy} is follows:

$$K_{RBF}(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \quad (7)$$

$$K_{Ploy}(x_i, x_j) = [(x_i \cdot x_j) + 1]^d \quad (8)$$

MBDA adopts $A - Distance$ to empirically estimate γ and the specific process is as follows:

- (1) Use SVM to train two-classifiers h to distinguish source data from target data and obtain the loss value $err(h)$;
- (2) Calculate $A - Distance$ between source data and target data, and the formula is as follows:

$$A(X_S, X_T) = 2(1 - 2err(h)) \quad (9)$$

- (3) Calculate the balance factor γ , and the formula is as follows:

$$\gamma = \frac{A_{Ploy}(X'_S, X'_T)}{A_{RBF}(X'_S, X'_T) + A_{Ploy}(X'_S, X'_T)} \quad (10)$$

where X'_S and X'_T respectively represent source data and target data after kernel mapping. $A_{RBF}(X'_S, X'_T)$ and $A_{Ploy}(X'_S, X'_T)$ represent the values of $A - Distance$. The larger the value is, the greater the difference between source data and target data after the kernel mapping, so the weight of the corresponding kernel function is smaller, and vice versa.

The balance factor μ plays an important role in minimizing the marginal and conditional distribution discrepancy between the domains. BDA evaluates the μ by searching its values in experiments, but it is not an effective solution. In order to effectively adjust the marginal and conditional distribution of the importance on different tasks, we estimate the balance factor μ by adopting $A - Distance$, and its formula is as follows:

$$\mu = \frac{A_{Marginal}(X'_S, X'_T)}{A_{Marginal}(X'_S, X'_T) + A_{Conditional}(X'_S, X'_T)} \quad (11)$$

where $A_{Marginal}(X'_S, X'_T)$ represents the $A - Distance$ value of the marginal probability distribution for source and target domains, and $A_{Conditional}(X'_S, X'_T)$ represents the $A - Distance$ value of the conditional probability distributions for the source and target domains. When $\mu \rightarrow 0$, it means that there is a big

difference between source data and target data. Therefore, the marginal distribution adaptation is more important, and vice versa.

2.3. Stacked Autoencoder Neural Network

2.3.1. Autoencoder

As one of the classic models of neural networks, the autoencoder consists of two stages of encoding and decoding. Its structure is shown in Figure 1. The autoencoder converts the original data of the high-dimensional space into the coding vector of the low-dimensional space by encoding, and then reconstructs the coding vector into the original data by decoding. The specific implementation process is as follows:

- Coding phase: the information is transmitted from front to back.

$$\begin{aligned} h_1 &= f(z_1) \\ z_1 &= w_1 x_1 + b_1 \end{aligned} \quad (12)$$

where, assuming the input layer is $X = \{x_1, x_2, \dots, x_n\}$, the subscript in the formula indicates that there are n training samples, w_1 and b_1 are the weight and bias of the encoding layer, respectively, and $f(*)$ is the excitation function, which is usually a sigmoid or tanh function.

- Decoding phase: the information is transmitted from the back to the front:

$$\begin{aligned} \hat{x}_1 &= f(z_2) \\ z_2 &= w_2 h_1 + b_2 \end{aligned} \quad (13)$$

where \hat{x}_1 is the output value of the decoding layer, w_2 and b_2 are the weights and bias of the decoding layer, respectively, and $w_2 = w_1^T$.

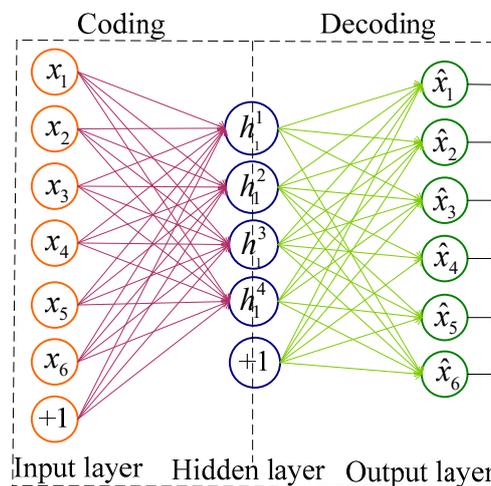


Figure 1. Autoencoder structure diagram.

As can be seen from the above process, the autoencoder belongs to unsupervised training. The training process of the autoencoder is to find the network parameters to minimize the reconstruction error on the training set D . The reconstruction error is generally the quadratic cost function, and the expression is as follows:

$$\begin{aligned} J_{AE} &= \sum_{x \in D} L(\hat{x}_1, x_1) \\ L(\hat{x}_1, x_1) &= \frac{1}{N} \|\hat{x}_1 - x_1\|^2 \end{aligned} \quad (14)$$

where \hat{X} is the predicted value, N is the number of samples, and L is the reconstructed error function. When the reconstruction error is small enough, the compressed feature vectors of the encoding layer retain most of the information of the original data.

2.3.2. Stack Autoencoder Neural Network

The SAE neural network is a deep neural network composed of multiple autoencoders. The basic principle is that the output of the previous autoencoder is used as the input of the latter autoencoder. Its structure is shown in Figure 2. Compared with the autoencoder network, the SAE neural network is more expressive and can extract more abundant features from original data. The encoding process of an m -layer SAE neural network is as follows:

$$\begin{aligned} a^{(l)} &= f(z^{(l)}) \\ z^{(l+1)} &= w^{(l,1)}a^{(l)} + b^{(l,1)} \end{aligned} \quad (15)$$

where $a^{(l)}$ is the output value of the l th encoding layer; $w^{(l,1)}, b^{(l,1)}$ is the weight and bias of the l th encoding layer; and $z^{(l)}$ and $z^{(l+1)}$ are the input value of the l th layer and the $l + 1$ th layer, respectively. The SAE neural network transmits information from the back to the front at the decoding process:

$$\begin{aligned} a^{(m+l)} &= f(z^{(m+l)}) \\ z^{(m+l+1)} &= w^{(m-l,2)}a^{(m+l)} + b^{(m-l,2)} \end{aligned} \quad (16)$$

where $a^{(m)}$ is the activation value of the deepest hidden unit, and $w^{(n-l,2)}, b^{(n-l,2)}$ are the weight and bias of the decoding layer. The SAE network structure is as follows:

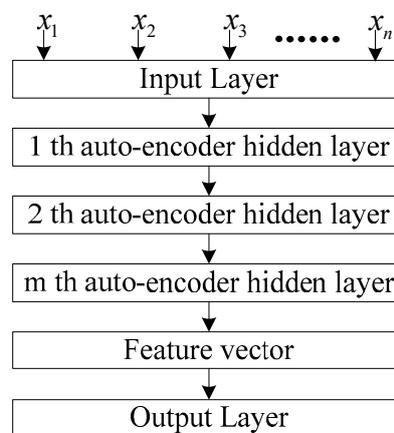


Figure 2. Stacked Autoencoder Neural Network.

The training process of the SAE neural network with classification function includes two stages: model pre-training and model fine-tuning:

- Model pre-training. The SAE neural network is constructed and the network model parameters are initialized through the unsupervised layer-by-layer training mode. Through pre-training, all hidden layers are obtained, and the features learned at each layer represent different levels of data characteristics.
- Model fine-tuning. Add a classification layer at the top of the SAE network and fine-tune the pre-training parameters to implement the classification function. Fine-tuning training takes all the layers of the SAE neural network as a whole model to train. At each iterative training, each parameter of the model is optimally adjusted. Therefore, fine-tuning training can improve the performance of SAE deep neural networks.

In this paper, the quadratic cost function and the cross-entropy cost function are used as the objective function of the first and second stages. The quadratic cost function is elaborated in Section 2.3.1, and the cross-entropy cost function is as follows:

$$z_i^j = \frac{e^{h_{\theta,j}(x_i)}}{\sum_{j=1}^C e^{h_{\theta,j}(x_i)}} \quad (17)$$

$$J[\theta] = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^C \{y_i = j\} \log(z_i^j) \quad (18)$$

where z_i^j is the predicted probability that the x_i belongs to category j , $h_{\theta,j}(x_i)$ represents the j th value of the output vector, and θ is neural network parameter. The SAE neural network can automatically extract features and has powerful feature expression capabilities. In fault diagnosis, the SAE neural network has the functions of noise reduction filtering and feature extraction.

3. Bearing State Recognition Method and Process under Different Working Conditions

The algorithm flow of the bearing state recognition method based on transfer learning under different working conditions is shown in Figure 3.

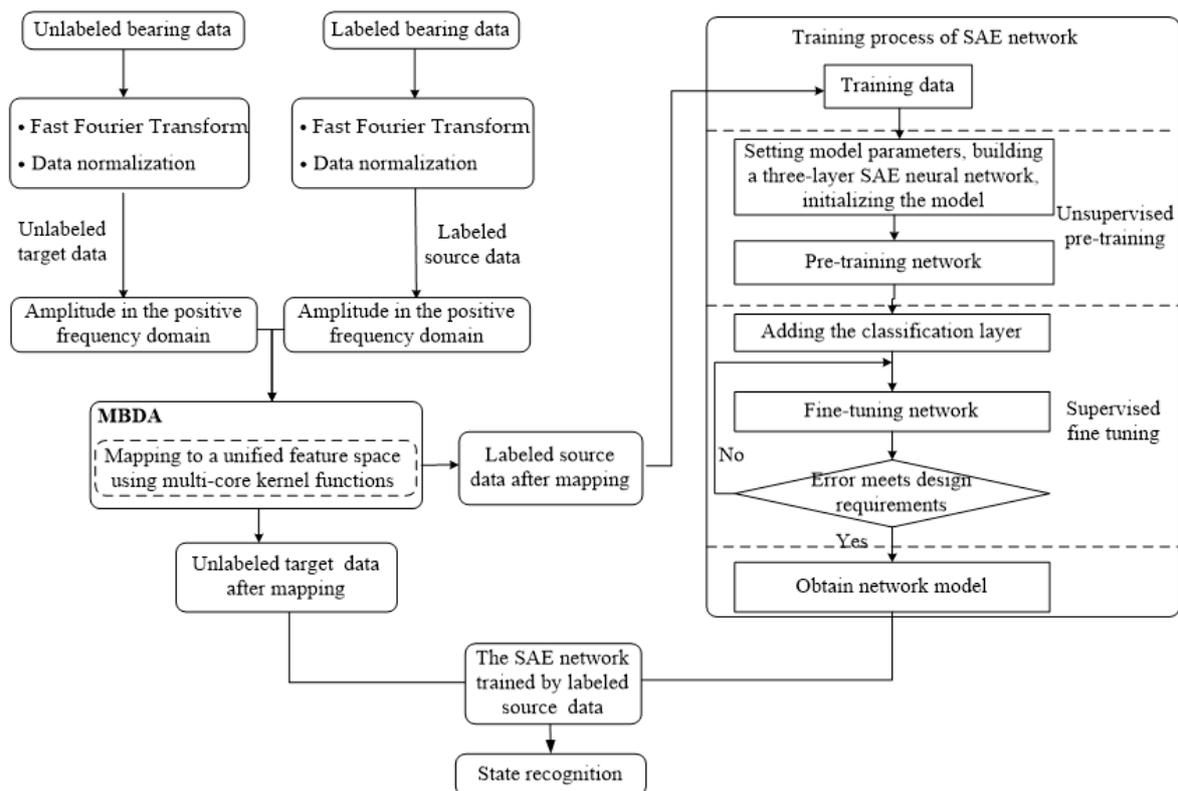


Figure 3. Flowchart of the bearing state recognition method under different working conditions.

The specific implementation process is as follows:

1. Calculate the spectrum of labeled bearing data and unlabeled bearing data, and normalize the amplitude to the range of $[0, 1]$. Because the signal's spectral amplitude is symmetrical about the origin, the positive frequency part is used as feature vectors. This not only ensures that information is not lost, but also reduces the number of calculations. The positive frequency

domain amplitude of labeled bearing data is used as labeled source data, and the positive frequency domain amplitude of unlabeled bearing data is used as unlabeled target data.

2. Map labeled source data and unlabeled target data in (1) to the same feature space by using the MBDA algorithm.
3. The training process of SAE is also a feature self-learning process, which can further extract features. The training process of SAE includes two parts: unsupervised pre-training and supervised fine tuning. Unsupervised pre-training is used to initialize network parameters, and supervised fine-tuning implements classification by adding a classification layer on top of the network. Labeled source data after spatial mapping in (2) are used as training samples to train the model, and finally the training model is obtained. Unlabeled source data after spatial mapping in (2) are input into the model, and the rolling bearing state recognition results are obtained.

4. Experimental Verification

4.1. Experimental Data

The experimental data are from the bearing data center of the Case Western Reserve University Laboratory. In this experiment, the SKF6205 drive end bearing vibration data is used as experimental data. Four different rotational speeds and different motor loads represent different working conditions A, B, C and D. Each working station includes four states: normal state (NS), internal raceway fault (IF), external raceway fault (OF) and ball fault (BF). The detailed information of the experimental data is shown in Table 1.

Table 1. Vibration signal parameter table for experimental data.

Different Working Conditions	RPM (r/min)	Motor Load (W)	Fault Diameter of IF, OF and BF (mm)	F _s (kHz)	Number of Samples
A	1730	2.25	0.1778	12	1500
B	1750	1.5	0.3356		1500
C	1772	0.75	0.5334		1500
D	1797	0	0.7112		1500

In Table 1, the fault diameter of IF, OF and BF indicates the diameter of a bearing inner raceway fault, outer raceway fault and ball fault. This paper constructs data sets under single and multiple working conditions, as shown in Table 2.

Table 2. Sample sets composition of rolling bearings under different working conditions.

Sample Sets	Source Data	Target Data	Source Data Sample Number	Target Data Sample Number
Single/single conditions	B	A	1500	1500
Single/multiple conditions	BC	A	3000	1500
Multiple/multiple conditions	CD	AB	3000	3000
Single/multiple conditions	BCD	A	4500	1500

Where A(T)-B(S) indicates that the source data (training data) is from working condition B, and the target data (test data) is from working condition A; A(T)-BC(S) indicates that source data (training data) is from working condition BC, and target data (test data) is from working condition A; AB(T)-CD(S) indicates that source data (training data) is from working condition CD, and target data (test data) is from working condition AB; A(T)-BCD(S) indicates that source data (training data) is from working condition BCD, and target data (test data) is from working condition A.

4.2. Model Performance Analysis

In the experiment, source data (training data) and target data (test data) are selected from single or multiple working conditions data sets. Taking A(T)-B(S) as an example, labeled source data B(S) is used to train the SAE neural network, and unlabeled target data A(T) is input into the model to obtain the bearing state. Training samples after fast Fourier transform (FFT) are normalized, and the amplitude of the positive frequency part is taken as a feature vector to train the SAE network. In the experiment, the normalized positive frequency part of the vibration data is used as the feature vector. This not only ensures that information is not lost, but also reduces the number of calculations. Generally, at neural network structure, the higher the number of network layers, the stronger the network expression ability. However, when the number of network layers is too large, it is difficult to train the model. After the previous experiments, this paper uses a three-layer SAE network model, the structure of the network is set to 64-32-16, bath_size is set to 100, and the number of iterations is 200. In the experiment, the normalized positive frequency part of the vibration data is used as the feature vector. This not only ensures that information is not lost, but also reduces the number of calculations. Under the four working conditions A(T)-B(S), A(T)-BC(S), AB (T)-CD (S) and A (T)-BCD (S), the curves of bearing state recognition accuracy of this method are shown in Figure 4.

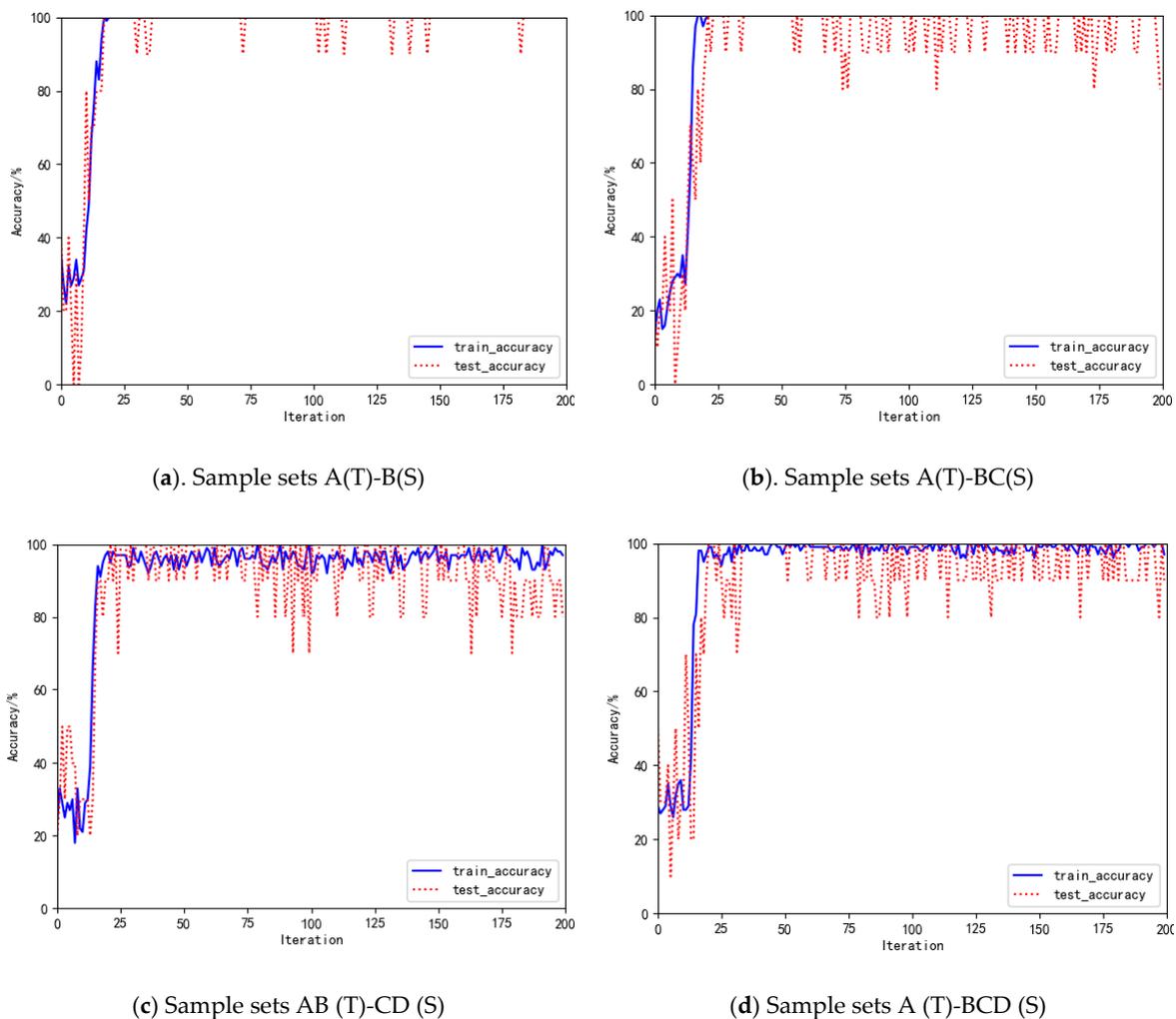


Figure 4. Bearing state recognition accuracy curve under working conditions A(T)-B(S), A(T)-BC(S), AB (T)-CD (S) and A (T)-BCD (S), respectively.

Figure 4a–d are graphs showing the state recognition accuracy of test data (target data) and training data (source data) under four working conditions, respectively. The trend of the accuracy of training data (source data) and test data (target data) is synchronized. Therefore, there is no over-fitting. In the case of single/single conditions A (T)-B (S), the state recognition accuracy of the test data almost reaches 100%. In the case of single/multiple conditions A (T)-BC(S) and A (T)-BCD(S), the state recognition accuracy of the test data is 98.50% and 96.86%, respectively. In the case of multiple/multiple conditions AB (T)-CD (S), the state recognition accuracy of the test data reaches almost 90.50%. The experimental results show that the MBDA-SAE method can obtain higher state recognition accuracy under variable working conditions.

4.3. Analysis and Comparison of MBDA and other Algorithms

In order to prove the advantages of the MBDA method, this paper introduces traditional transfer learning methods such as the TCA, JDA and BDA methods. The results after different transfer learning methods and SAE networks under variable working conditions are shown in Table 3. As can be seen from the above table, the state recognition accuracy of the MBDA-SAE method is higher than that of the TCA-SAE, JDA-SAE, and BDA-SAE methods. The reason is that the multi-core kernel function has a good advantage in dealing with the imbalance between source data and target data. JDA-SAE and BDA-SAE methods take the difference of marginal distribution and conditional distribution as objective functions, so their state accuracy is higher than the TCA-SAE method. Compared to the JDA-SAE method, the BDA method adds a balance factor obtained by searching in experiments to balance the marginal distribution and conditional distribution. The BDA-SAE method achieves higher state recognition accuracy at the expense of efficiency.

Table 3. Bearing state recognition accuracy under different methods (%).

Different Methods/Sample Sets	A(T)-B(S)	A(T)-BC(S)	AB(T)-CD(S)	A(T)-BCD(S)	Average Accuracy
TCA-SAE	75.00	69.00	62.00	54.00	65.00
JDA-SAE	92.00	77.00	69.52	69.00	76.88
BDA-SAE	96.99	88.00	83.10	77.00	86.27
MBDA-SAE	100.00	98.50	96.86	90.50	96.47

5. Conclusions

In this paper, we propose a rolling bearing state recognition method based on a WDBA-SAE neural network under different working condition data. The following conclusions have been obtained through experiments:

- (1) This method depends on BDA theory, and constructs a weighted mixed kernel function to map different working condition data to a unified feature space, which effectively minimizes the distribution divergence between different working conditions data. The MDBA method does not need to obtain the cross-characteristics of different working conditions data in advance, which simplifies data processing.
- (2) This paper adopts the A – Distance algorithm to calculate the balance factor of the distribution and the balance factor of the kernel function. It can adaptively balance the importance of the marginal and conditional distribution and the importance of different kernel functions, and improve efficiency.
- (3) The MDBA method was compared to other transfer learning methods, such as TCA, JDA and BDA. In the case of a single/single condition A (T)-B (S), the accuracy of the bearing state recognition methods based on the JDA-SAE, BDA-SAE and MDBA-SAE methods reached more than 90%. However, the diagnostic accuracy based on the TCA-SAE method is 75%. In the case of multiple/multiple conditions AB (T)-CD (S), the state recognition accuracy of the method

proposed in this paper reaches more than 90%. However, the accuracy of other methods is less than 80%. Therefore, the advantages of this method are more obvious under multiple/multiple conditions. Experiments showed that the MDBA method can better recognize the unknown state of rolling bearings under variable working conditions.

The follow-up work of this article is as follows:

- (1) During the deep neural network training process, multiple experiments are required to determine better hyperparameters (such as the number of network layers, the number of neurons, the number of iterations, etc.), and then the setting of the hyperparameters will be studied;
- (2) The features extracted from the multi-layer network feature space will be visualized;
- (3) This article only studies bearing-related faults, and subsequent studies will distinguish other faults, such as unbalanced loads and broken rotor bars.

Author Contributions: B.C. conducted data analysis. N.C. designed the research, validated the calculations and wrote the original draft. Z.J. and J.G. revised the original draft and supervised the work. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the 13th Five-Year Pre-Research Project of the Equipment Development Department grant number 41404030202.

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

References

1. Sun, H.; He, Z.; Zi, Y.; Yuan, J.; Wang, X.; Chen, J.; He, S. Multiwavelet transform and its applications in mechanical fault diagnosis—A review. *Mech. Syst. Signal Process.* **2014**, *43*, 1–24. [\[CrossRef\]](#)
2. Tse, P.W.; Peng, Y.H.; Yam, R. Wavelet analysis and envelope detection for rolling element bearing fault diagnosis - their effectiveness and flexibilities. *J. Vib. Acoust.* **2001**, *123*, 303–310. [\[CrossRef\]](#)
3. Žvokelj, M.; Zupan, S.; Prebil, I. EEMD-based multiscale ICA method for slewing bearing fault detection and diagnosis. *J. Sound Vib.* **2016**, *370*, 394–423. [\[CrossRef\]](#)
4. Ma, B.; Jiang, Z.N. Envelope analysis based on Hilbert transform and its application in rolling bearing fault diagnosis. *J. B. Univ. Chem. Technol.* **2004**, *31*, 36–39.
5. Xu, Y.; Meng, Z.; Zhao, G. Study on compound fault diagnosis of rolling bearing based on dual-tree complex wavelet transform. *Chin. J. Sci. Instrum.* **2014**, *35*, 447–452.
6. Ren, Z.; Zhou, S.; Chunhui, E.; Gong, M.; Li, B.; Wen, B. Crack fault diagnosis of rotor systems using wavelet transforms. *Comput. Electr. Eng.* **2015**, *45*, 33–41. [\[CrossRef\]](#)
7. Chen, X.H.; Cheng, G.; Shan, X.L.; Hu, X.; Guo, Q.; Liu, H.G. Research of weak fault feature information extraction of planetary gear based on ensemble empirical mode decomposition and adaptive stochastic resonance. *Measurement* **2015**, *73*, 55–67. [\[CrossRef\]](#)
8. Guo, L.; Gao, H.; Zhang, Y.; Huang, H. Research on Bearing State Recognition Based on Deep Learning Theory. *J. Vib. Shock* **2016**, *35*, 167–171.
9. Kilundu, B.; Letot, C.; Dehombreux, P.; Chimentin, X. Early Detection of Bearing Damage by Means of Decision Trees. *IFAC Proc. Vol.* **2008**, *41*, 211–215. [\[CrossRef\]](#)
10. Wu, S.D.; Wu, C.W.; Wu, P.H.; Ding, J.J.; Wang, C.C. Bearing fault diagnosis based on multiscale permutation entropy and support vector machine. *Entropy* **2012**, *14*, 1343–1356. [\[CrossRef\]](#)
11. Satish, B.; Sarma, N.D.R. A fuzzy bp approach for diagnosis and prognosis of bearing faults in induction motors. In Proceedings of the IEEE Power Engineering Society General Meeting (PESGM), San Francisco, CA, USA, 12–16 June 2005; pp. 2291–2294.
12. Li, H.; Zhang, H.; Qin, X.R.; Sun, Y.T. Fault diagnosis method for rolling bearings based on short-time Fourier transform and convolution neural network. *J. Vib. Shock* **2018**, *37*, 124–131.
13. Wang, F.; Jiang, H.; Shao, H. An adaptive deep convolutional neural network for rolling bearing fault diagnosis. *Meas. Sci. Technol.* **2017**, *28*, 095005.
14. Wang, F.; Liu, X.; Deng, G.; Yu, X.; Li, H.; Han, Q. Remaining Life Prediction Method for Rolling Bearing Based on the Long Short-Term Memory Network. *Neural. Process Lett.* **2019**, *10*, 1–18. [\[CrossRef\]](#)

15. Hinch, A.Z.; Tkiouat, M. Rolling element bearing remaining useful life estimation based on a convolutional long-short-term memory network. *Procedia. Comput. Sci.* **2018**, *127*, 123–132. [[CrossRef](#)]
16. Shao, H.; Jiang, H.; Wang, F.; Wang, Y. Rolling bearing fault diagnosis using adaptive deep belief network with dual-tree complex wavelet packet. *ISA Trans.* **2016**, *69*, 187–201. [[CrossRef](#)] [[PubMed](#)]
17. Shao, H.; Jiang, H.; Zhang, X.; Niu, M. Rolling bearing fault diagnosis using an optimization deep belief network. *Meas. Sci. Technol.* **2015**, *26*, 115002. [[CrossRef](#)]
18. Chen, R.X.; Yang, X.; Yang, L.X. Fault severity diagnosis method for rolling bearings based on a stacked sparse denoising auto-encoder. *J. Vib. Shock* **2017**, *36*, 125–131+137.
19. Duan, L.; Tsang, I.W.; Xu, D. Domain transfer multiple kernel learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 465–479. [[CrossRef](#)]
20. Pan, S.J.; Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **2009**, *22*, 1345–1359. [[CrossRef](#)]
21. Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How transferable are features in deep neural networks? *Adv. Neural Inf. Process. Syst.* **2014**, *27*, 3320–3328.
22. Pan, S.J.; Tsang, I.W.; Kwok, J.T.; Yang, Q. Domain Adaptation via Transfer Component Analysis. *IEEE Trans. Neural Netw.* **2011**, *22*, 199–210. [[CrossRef](#)] [[PubMed](#)]
23. Wang, J.; Chen, Y.; Hao, S.; Feng, W.; Shen, Z. Balanced Distribution Adaptation for Transfer Learning. In Proceedings of the 2017 IEEE International Conference on Data Mining (ICDM), New Orleans, LA, USA, 18–21 November; pp. 1129–1134.
24. Long, M.; Wang, J.; Ding, G.; Sun, J.; Yu, P.S. Transfer Feature Learning with Joint Distribution Adaptation. In Proceedings of the 2013 IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 2200–2207.
25. Kang, S.Q.; Hu, M.W.; Wang, Y.J.; Xie, J.; Mikulovich, V.I. Fault Diagnosis Method of a Rolling Bearing Under Variable Working Conditions Based on Feature Transfer Learning. *Proc. CSEE* **2019**, *39*, 764–772.
26. Fernando, B.; Habrard, A.; Sebban, M.; Tuytelaars, T. Unsupervised visual domain adaptation using subspace alignment. In Proceedings of the IEEE international conference on computer vision (ICCV 2013), Sydney, Australia, 1–8 December 2013; pp. 2960–2967.
27. Chen, C.; Shen, F.; Yan, R. Enhanced least squares support vector machine-based transfer learning strategy for bearing fault diagnosis. *Chin. J. Sci. Instrum.* **2017**, *38*, 33–40.



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).