



Article Transfer Learning Strategies for Deep Learning-based PHM Algorithms

Fan Yang ^{1,2}, Wenjin Zhang ¹, Laifa Tao ^{1,2}, and Jian Ma ^{1,2,*}

- ¹ School of Reliability and Systems Engineering, Beihang University, Beijing 100191, China; maowuxin1900@163.com (F.Y.); zwjok@buaa.edu.cn (W.Z.); taolaifa@buaa.edu.cn (L.T.)
- ² Science & Technology Laboratory on Reliability & Environmental Engineering, Beijing 100191, China
- * Correspondence: 09977@buaa.edu.cn; Tel.: +86-15910634863

Received: 18 February 2020; Accepted: 27 March 2020; Published: 30 March 2020



Featured Application: The transfer strategies proposed in this paper can guide the industry to reuse and develop the existing PHM algorithm based on deep learning under different data conditions and save the cost of algorithm development and data collection to the greatest extent.

Abstract: As we enter the era of big data, we have to face big data generated by industrial systems that are massive, diverse, high-speed, and variability. In order to effectively deal with big data possessing these characteristics, deep learning technology has been widely used. However, the existing methods require great human involvement that is heavily depend on domain expertise and may thus be non-representative and biased from task to similar task, so for a wide variety of prognostic and health management (PHM) tasks, how to apply the developed deep learning algorithms to similar tasks to reduce the amount of development and data collection costs has become an urgent problem. Based on the idea of transfer learning and the structures of deep learning PHM algorithms, this paper proposes two transfer strategies via transferring different elements of deep learning PHM algorithms, analyzes the possible transfer scenarios in practical application, and proposes transfer strategies applicable in each scenario. At the end of this paper, the deep learning algorithm of bearing fault diagnosis based on convolutional neural networks (CNN) is transferred based on the proposed method, which was carried out under different working conditions and for different objects, respectively. The experiments verify the value and effectiveness of the proposed method and give the best choice of transfer strategy.

Keywords: transfer learning; transfer strategies; deep learning; bearing; fault diagnosis

1. Introduction

Prognostics and health management (PHM) is a framework that offers comprehensive yet individualized solutions for managing system health [1], including prognostics and health management. Prognostics is usually defined as the generation of current system status evaluation results through fault data or historical maintenance data collected by sensors, combined with the future working environment of the system to predict the system's remaining useful life (RUL) or end of life (EOL); health management is defined as the process of measuring, recording and monitoring the extent to which the system deviates from normal operating conditions in real time. As an engineering discipline PHM aims to provide users with an integrated view of the health state of a machine or an overall system. Diagnostics are also included in prognostics and health management [2].

In today's increasingly complex industrial systems, PHM technology provides important support for system design and development, production and manufacturing, operation, logistics support and maintenance, decommissioning and destruction. Because the big data generated by industrial systems is massive, diverse, high-speed, and variable, deep learning technology has been added to overcome the existing problems of prognostics and diagnostics. The deep learning-based PHM technology has been used in fault diagnosis and health evaluation of motors, gearboxes, bearings, etc., and has achieved better results than traditional methods [3–8].

The PHM algorithm is the core of PHM technology. It provides the data processing foundation for fault diagnostics, prognostics and health management, covering the entire process from data preprocessing, condition monitoring, fault diagnostics, prognostics to health management. The development and application of PHM algorithms is mostly based on proprietary systems, and deep learning PHM algorithms are no exception. This means that the deep learning PHM algorithms developed based on a certain instance are difficult to effectively reuse and generalize. For similar problems, rebuilding the algorithm from scratch means a lot of manpower, material resources and time. At the same time, if similar tasks do not have enough data support, the newly developed algorithms will have difficulty in displaying good task performance or may even fail to complete tasks.

Taking rotary bearing fault diagnosis as an example, the training data of a fault diagnosis classifier may come conditions where the motor has no load, while the motor may have different load states in practical application. Although the failure mode has not changed, the target data distribution of different load states has changed, so the classifier trained by the training data (source domain) cannot be directly used for the target data (target domain). It is undoubtedly expensive to collect enough fault data available for training to reconstruct the fault diagnosis classifier, and it is difficult to directly train a satisfactory fault diagnosis classifier for the collection of a small amount of data [9]. In addition to the above cases, if the rotary bearing is replaced, the original fault diagnosis classifier will not be applicable. Even if sufficient target domain data is collected, the data reprocessing and network construction tuning will be very expensive and inefficient.

To solve this challenge, a transfer learning technique, whose major research focus is how to store the knowledge gained when solving a problem and apply it to different but related problems, would be needed and desirable to reduce the amount of development and data collection costs and improve the performance of algorithms in the target domain. There are already many examples in knowledge engineering where transfer learning can be truly promising and useful, including object recognition, image classification and natural language processing [10–12].

Lu [13] proposed a deep neural network for fault diagnosis-based domain adaption in 2017. Combined with the idea of transfer learning, the problem of label-free transfer in the target domain was solved. Xie [14] achieved feature transfer based on the transfer component analysis (TCA) method in gearbox fault diagnosis. Although the research on transfer of fault diagnosis algorithms is gradually increasing [15–17], most of them are about unlabeled transfer learning and there is litter research focus on the solution for the situation that the amount of data in the target domain is small or the similarity between the target domain and the source domain is low. In order to solve the above problems, we propose a deep learning PHM algorithm transfer technology combined with the idea of transfer learning, including two transfer strategies and the analysis of applicable transfer scenarios.

The rest of this paper is organized as follows: In Section 2, the background information of transfer learning is outlined and the transfer scenarios are defined according to the data situation of the target domain and the source domain. In Section 3, our transfer strategies are proposed and which strategy should be applied in which scenario are talked. In Section 4, the application effect of different transfer strategies in different transfer scenarios is tested and verified. Section 5 summarizes and gives suggestions for the application of the method proposed in the industry.

Notations which are used in this work frequently are summarized in Table 1.

Notation	Description	Notation	Description
D_s	Source Domain	D_t	Target Domain
S	Transfer Scenario	TS	Transfer Strategy
TM	Transfer Method	A1	Algorithm 1

Table 1. Notations and Desci	iptions.
------------------------------	----------

2. Deep Learning PHM Algorithm Transfer Scenario

When transferring a deep learning PHM algorithm, it is necessary to specify the transfer scenario. The similarity between the target domain and the source domain and the amount of data in the target domain are important indicators for the choice of transfer method. Therefore, according to the similarity of data and the size of data set in the target domain, the transfer scenarios of deep learning PHM algorithm are divided into four categories. Figure 1 shows the four types of scenarios for deep learning PHM algorithm transfer.



Size of data set

Figure 1. Deep learning PHM algorithm transfer scenario map.

In scenario I (S1), the target domain data set is large and has a high degree of similarity with the source domain data, which means that the weights of network obtained from the training of the source domain data can be retained as far as possible. On this basis, using the target domain data for very few batches of training can obtain a satisfactory deep learning PHM algorithm in the target domain.

The high similarity between target domain data and source domain data in scenario II (S2) makes the transfer process similar to S1, but fewer target domain data require more batches of training in order to obtain a satisfactory deep learning PHM algorithm for the target domain. The fault diagnosis task of the same rotating bearing under different loads can be regarded as a concrete expression of S1 and S2. According to the amount of data collected under different loads, the S1 or S2 can be specifically distinguished.

In scenario III (S3), the data set of target domain is small so that it is difficult to achieve the desired diagnostic goal if the network trained directly in the target domain. We transfer as many weights of the network trained in the source domain as possible and fine-tune with a small amount of data in the target domain to let the new network acquire more information of source domain easily so that the desired diagnostic goal can more probably be achieved.

In scenario IV (S4), the target domain data set is large enough. In this scenario, continuing to use the network weights gained in source domain may make the whole network weights fall into a local optimum. For this scenario, we only transfer the network structure, and initialization weights retraining will be more reasonable. Fault diagnosis tasks of different types of rotary bearings under the same working condition can be considered as the concrete manifestation of S3 and S4. According to the amount of data collected in the target domain, S3 or S4 can be specifically distinguished.

3. Transfer Strategies of Deep Learning PHM Algorithms

The idea of transfer learning is to let a new algorithm inherit the knowledge (knowledge here represents the ability of existing algorithms to extract and analyze data features from the source domain) of the existing algorithm. Just as the teacher teaches the student knowledge, the higher level of summary knowledge transfer is undoubtedly the fastest and most efficient. Deep neural networks are based on network weights that come from information obtained from large amounts of data. If the weights in the existing neural network are extracted and transferred to the new neural network, it means that we "transfer" the learned features without having to retrain a network from scratch.

Based on this idea, we regard the existing deep learning PHM algorithm as a pre-training model. When encountering a portable task, we transfer the network structure or weights of the existing algorithm and obtain a new deep learning PMM algorithm needed by the new task through very little training.

The object in the transfer process has two parts: network structure and weights. According to whether the network weights are transferred during the transfer process, this paper proposes two types of deep learning PHM algorithm transfer strategy:

TS1: Transfer both structure and weights of network simultaneously.

TS2: Transfer only structure of network and initialize weights of network.

The implementation process of the two strategies is shown in Figures 2 and 3.



Figure 2. TS1 Transfer both structure and weights of network simultaneously.

TS1 transfers the structure and weights of the network at the same time, so that the network in the target domain inherits more source domain information, which is suitable for scenarios where the target domain and source domain are similar. TS2 only transfers the network structure. Its weights need to be retrained using the target domain data after initialization. It inherits the network structure design experience of the source domain, but the inheritance in the knowledge layer is relatively small, which is more suitable for the scenarios where the target domain task is similar to the source domain but the sample similarity is low.



Figure 3. TS2 Transfer only the network structure, the weights of the new network is randomly initialized.

These two types of transfer strategies can be used alone or in combination. For example, when transferring a diagnosis fault diagnosis algorithm based on a CNN network, three transfer methods can be designed according to the transfer tuple. When a CNN network is used for fault diagnosis, its structure generally consists of convolutional layers, pooling layers, fully connected layers, and a final classification layer. Among them, convolutional layers and pooling layers need to be connected sequentially, fully connected layers can be flexibly modified, and the classification layer needs to be placed at the end of the entire network structure. Convolution layers, pooling layers and full connection layers are mainly used for data feature extraction and compression. Transfer of these network layers can inherit the feature extraction and compression mode of source domain; classification layer is used to achieve accurate fault mode diagnosis, and different classification layers should be designed for different fault diagnosis tasks. Different layer types of transfer can produce different transfer effects. Therefore, based on TS1 and TS2, three transfer methods can be designed for CNN network used for fault diagnosis:

TM1: Combining TS1 and TS2. Transfer the network structure of the convolutional and pooling layers of the source domain and their weights. The fully connected layer and classification layer only transfer the network structure, and then use the target domain data for retraining.

TM2: TS1. Transfer all network structures and network weights in the source domain, and then use the target domain data for retraining.

TM3: TS2. Transfer the source domain network structure, initialize the weights, and then use the target domain data for retraining.

The application scenarios of the three transfer methods are analyzed. In S1 and S2, the target domain and source domain samples have a high degree of similarity. Inheriting as much of the source domain information as possible will be more helpful for the task of the target domain, so we can use TM1 or TM2; in S3 The target domain and source domain samples have a low similarity, but because the amount of data in the target domain is small, it is difficult to support the effective training of the target domain algorithm. Therefore, we should also inherit as much source information as possible and TM1 or TM2 can be used. In S4, the target domain and source domain samples have a low similarity, but the sample size of the target domain is sufficient to support the effective training of the target domain algorithm. Therefore, the structural design experience of the source domain algorithm should be inherited to reduce the algorithm structure design cost. At the same time, the target domain data is used for complete retraining to avoid negative interference caused by the source domain data. TM3 can be used.

4. Case, Transfer of Rotating Machinery Fault Diagnosis Algorithm

4.1. Data Introduction

The bearing data of Case Western Reserve University are used. The bearing test device is shown in Figure 4. On the left is a two-horsepower three-phase induction motor (model: 2HP IQPreAlert Reliance Electric, Milwaukee, Wisconsin, US), on the right are a power meter and an electronic control unit (not shown), which are connected by a middle torque sensor. In the test, the acceleration sensor is installed via magnetic mount on the drive end at 12 o'clock, the fan end at 12 o'clock and the motor support chassis respectively. The test data is collected by the acceleration sensor and is a vibration signal that reflects the state of the bearing.



Figure 4. Bearing test device.

The test bearings support the motor shaft, and single point faults are arranged on the inner ring, outer ring (6:00, 3:00 and 12:00 directions) and rolling element of the bearings respectively by EDM technology. The diameter of faults includes 0.007, 0.014, 0.021, 0.028 and 0.040 inches (1 inch = 2.54 cm). Vibration data were collected at 0, 1, 2 and 3 horsepower, and the frequency was 12 kHz. SKF bearings are used for 0.007, 0.014 and 0.021 fault diameters, and equivalent NTN bearings are used for 0.028 and 0.040 fault diameters.

Drive end bearing and fan end bearing are two kinds of bearings. The drive end bearing adopts 6205-2RS JEM, Z=9, double-sided seal deep groove ball bearings (SKF, Gothenburg, Sweden). The specific size is shown in Table 2.

Table 2.	Drive end	bearing	size	(inch).
----------	-----------	---------	------	---------

Inside Diameter	Outside Diameter	Thickness	Ball Diameter	Pitch Diameter
0.9843	2.0472	0.5906	0.3126	1.537

The fan end bearing adopts a SKF deep groove ball bearing 6203-2RS JEM, Z = 8, double-sided seal. The specific size is shown in Table 3.

fuble 5. I all clic bearing bize (incli).	Table 3.	Fan	end	bearing	size	(inch).
---	----------	-----	-----	---------	------	---------

Inside Diameter	Outside Diameter	Thickness	Ball Diameter	Pitch Diameter
0.6693	1.5748	0.4724	0.2656	1.122

In this paper, the bearing fault diagnosis algorithm A1 is developed via using the vibration data of the bearing fault at the drive end under 0 horsepower loads. Then the obtained fault diagnosis algorithm A1 is transferred to the vibration data of the bearing fault at the drive end under 2 horsepower loads to verify the transfer effect of the algorithm under different working conditions. Finally, A1 is transferred to vibration data of fan end bearing fault under 0 horsepower loads to verify the transfer effect of searing fault under 0 horsepower loads to verify the transfer effect of fan end bearing fault under 0 horsepower loads to verify the transfer effect of algorithm between different objects.

4.2. Development of Rotating Machinery Fault Diagnosis Algorithm

In order to verify the proposed deep learning PHM algorithm transfer strategies, the CNN-based bearing fault diagnosis algorithm was designed and developed.

4.2.1. Algorithm Design

(1) Data preprocessing

The data collected by Case Western Storage University are in MATLAB format (. mat). Each file contains the vibration data of fan end, bearing end and supporting chassis and the speed of motor. We import all the data into Python and store it in a multidimensional array.

The data preprocessing of the algorithm includes two parts: data enhancement and standardization. In order to satisfy the amount of data required for the deep learning algorithm training, data enhancement processing is performed on the existing fault data, and sufficient training data is obtained. The fault of the bearing will be reflected in every revolution. Therefore, it is considered that the data collected by the rotation of the motor can record the complete fault mode complete information, and the minimum data length carrying the complete fault mode information is:

$$L = \frac{F \times 60}{RPM_{min\,min}} \tag{1}$$

where *F* is the acquisition frequency and RPM_{min} is the minimum speed in the same batch of data. Segmentation is carried out every L_{min} data points, and the training data of the k-th fault mode is:

$$N_k = \frac{n_k - L_{min}}{L_{min}} \tag{2}$$

where n_k is the total amount of data collected by the *k*th fault, and L_{min} is the minimum length of data.

Through the above data enhancement method, more training test data can be obtained.

Data standardization is based on zero mean and variance normalization. The formula is as follows:

$$z = \frac{x - \mu}{\sigma} \tag{3}$$

where μ and σ are the mean and variance of the original data set respectively.

Through this standardization, every dimension of data can be subjected to zero-mean and one-variance normal distribution, so that each dimension of data feature dimension is equivalent. In this way, when clustering analysis of neural networks, every dimension is de-dimensionalized, which can avoid the influence of different dimension selection on neural network training.

(2) Fault diagnosis network based on CNN network structure

Compared with the multilayer perceptron (MLP) network, CNN has more prior knowledge of spatial correlation between data. Using a CNN network, three kinds of fault characteristic data can be observed simultaneously, and three kinds of fault feature data can be further combined to make fault diagnosis. The input of CNN can be a multi-dimensional structure, so the three-dimensional vibration data of driver end, fan end and support chassis after data preprocessing are directly sent into the CNN network in three-dimensional form for training.

CNN network can extract fault features from original data and classify faults. The number of layers and the number of nodes in each layer are selected based on previous experience on neural

network design and cross-checked on the data set. The structure of CNN network is shown in Table 4. It is a more general neural network structure for fault diagnosis. And in order to learn the relationship between two-dimensional vibration data as much as possible, a 2×2 convolution kernel is used.

Layer (type)	Output Shape	Param#
Input_1 (InputLayer)	(None, 3, 302, 1)	0
Conv2d_1 (Conv2D)	(None, 2, 301, 32)	160
Max_pooling2d_1 (MaxPooling2)	(None, 1, 150, 32)	0
Conv2d_2 (Conv2D)	(None, 1, 149, 64)	4160
Max_pooling2d_2 (MaxPooling2)	(None, 1, 74, 64)	0
Dropout_1 (Dropout)	(None, 1, 74, 64)	0
Flatten_1 (Flatten)	(None, 4736)	0
Dense_1 (Dense)	(None, 1024)	4850688
Dropout_2 (Dropout)	(None, 1024)	0
Dense_2 (Dense)	(None, 512)	524800
Dropout_3 (Dropout)	(None, 512)	0
Dense_3 (Dense)	(None, 64)	32832
Dropout_4 (Dropout)	(None, 64)	0
Dense_4 (Dense)	(None, 4)	260
Total params: 5412965		
Trainable params: 5412965		
None-trainable params: 0		

Table 4. Graph of network structure.

(3) Softmax classification layer

Because the five failure modes of bearings are mutually exclusive, the Softmax regression model is used to achieve multi-classification. The last layer of CNN network structure connects the Softmax classification layer. The function of Softmax is:

$$P(y = i | x; \theta) = \frac{\exp(\theta_i^T x)}{\sum_{k=1}^k \exp(\theta_k^T x)}$$
(4)

where x is the model input, $\theta^1, \theta^2, \dots, \theta^k \in \mathfrak{R}^n$ (The dimension of the feature vector x is n).

For training sets $\{(x^{(1)}, y^{(1)}), \dots, (x^{(m)}, y^{(m)})\}$, $y^{(i)} \in \{1, 2, \dots, 5\}$, there are five categories in total. For each input x there will be a probability $P(y = j|x; \theta)$ for each class. Softmax's cost function is defined as follows, which contains the indicator function 1{equation=True}=1.

$$J(\theta) = -\frac{1}{m} \left[\sum_{i=1}^{m} \sum_{j=1}^{k} 1\{y^{i} = j\} log(p(y^{(i)} = j | x^{(i)}; \theta)) \right]$$
(5)

We use the adaptive moment estimation (Adam) optimization algorithm to minimize the cost function of Softmax. The parameters of it are set as: learning_rate=0.001, beta1=0.9, beta2=0.999, epsilon=1e-08.

After derivation, the gradient formula is obtained as follows:

$$\nabla_{\theta_j} J(\theta) = -\frac{1}{m} \sum_{i=1}^m \left[x^{(i)} \left(\mathbf{1} \{ y^{(i)} = j \} - p \left(y^{(i)} = j | x^{(i)}; \theta \right) \right) \right]$$
(6)

Adam uses the first-order moment estimation and second-order moment estimation of the gradient to dynamically adjust the learning rate of each parameter. Adam has a certain range of learning rate per iteration, which makes the parameters more stable, and requires less memory, and calculates different adaptive learning rates for different parameters.

4.2.2. Result Analysis

The minimum data length $L_{min} = 400$ carrying the complete failure mode information is calculated by the formula (1). The training data amount of each failure mode is 302 calculated by the Equation (2). Figure 5 shows the process of data enhancement.



Figure 5. Graph of data preprocessing results.

The data obtained by the data enhancement is mapped by Equation (3) to finally obtain the training data for the neural network training, and the data is standardized as shown in Figure 5.

Then the training data are sent into the CNN network structure for training. The training iterations are 50 times and the batch size is 100. The training results are shown in Figure 6.



Figure 6. Training accuracy, training loss, test accuracy and test loss of CNN for bearing diagnosis.

4.3. Transfer of Fault Diagnosis Algorithm for Rotating Machinery under Different Working Conditions

The vibration diagnosis data of the fault of the drive end bearing under 0 horsepower and 2 horsepower are used to transfer the fault diagnosis algorithm under different working conditions. The source domain is the drive end bearing of 0 horsepower, and the target domain is the drive end bearing of 2 horsepower.

According to the ratio of the target domain of the source domain data, the data is divided into four groups. The target domain data size is 0.125 (this means the percent of used data and full data), 0.375, 0.625, and 0.875, respectively. We define a data set that is less than 50% of the total data as a small data set and a data set that is greater than 50% as a large data set. The impact of different target domain data size and transfer methods proposed in Section 3 on the transfer results was verified. The transfer process is shown in Figure 7. In TM1, the network weights are fully multiplexed from the source domain network and not need initializing. In TM2, the weights of the classification layer are initialized via glorot_uniform, and the remaining layers reuse the source domain network. In TM3, all network weights are via glorot_uniform.



Figure 7. Flowchart of bearing fault diagnosis algorithm transfer.

When the same bearing works under different working conditions, its mechanical structure has not changed, which means that the similarity between the source and target domains is high. Therefore, Transfer under different working conditions can be divided into S1 and S2 according to the corresponding target domain data size.

Figures 8–10 show the training accuracy, training loss, test accuracy and test loss for TM1, TM2, and TM3 for the four groups of experiments. From the changes in training accuracy and training loss, it can be seen that the size of data in the target domain has a small effect on the final fitting degree of the training, but the bigger the size of data, the faster the diagnostic algorithm converges and the shorter the time it takes to obtain a higher training accuracy. From the changes in test accuracy and test loss, it can be seen that when the target and source domains are similar, TM1 and TM2 are not sensitive to the size of data in the target domain, and TM3 is sensitive to the data in the target domain. The larger the size of data, the better the algorithm's final fault diagnosis effect.



Figure 8. Results of TM1 in different working conditions.



Figure 9. Results of TM2 in different working conditions.



Figure 10. Results of TM3 in different working conditions.

Figures 11–14 show the comparison of transfer methods in the four groups of experiments. It can be seen that TM3 in four target domain data quantities is not as good as the other two transfer methods in terms of fitting speed, convergence speed and test accuracy. When the size of data is small, the performance of TM1 is similar to that of TM2. With the increase of the amount of data, transfer TM2 is slightly better than TM1.



Figure 11. Comparison of three transfer methods under 0.125 data size.



Figure 12. Cont.



Figure 12. Comparison of three transfer methods under 0.375 data size.



Figure 13. Comparison of three transfer methods under 0.625 data size.



Figure 14. Comparison of three transfer methods under 0.875 data size.

The test accuracy of the algorithm is a performance indicator when the algorithm is finally applied to actual tasks. Table 5 summarizes the test accuracy of the four groups of experiments. The experiments related to TM1 are performed only once, because the network parameters used by it are completely inherited from the source domain network, and the experimental results are fixed. The experiments related to TM2 and TM3 were repeated five times, and the results shown in the table are the average results of repeated experiments. The reason why no more repeated experiments were

taken was that the results of the five experiments were almost the same, and the fluctuation was very small. Comparing the transfer methods, it can be seen that in both S1 and S2, the performance of TM1 and TM2 are higher than TM3, and the performance of TM2 is slightly higher than TM1. That is, as long as the source domain similarity of the target domain is high, as much as possible transfer the network structure and weights of the source domain and inheriting more source domain information can maximize the algorithm's fault diagnosis performance in the target domain. From the perspective of data size, it can be seen that under each transfer method, increasing the target domain data size improves the algorithm performance, but TM2's algorithm performance in 0.875 data size is almost equivalent to 0.125 data size. Using TM2 comes from TS1 for target domain fault diagnosis algorithm development can effectively save algorithm development costs and data acquisition costs in S1 or S2.

Transfer Scenario	Data Ration	Transfer Method 1	Transfer Method 2	Transfer Method 3
01	0.125	0.9767	9.9992	0.7600
SI	0.375	0.9855	0.9952	0.9759
63	0.625	0.9986	1.0000	0.9769
S2	0.875	0.9978	1.0000	0.9910

Table 5. Summary of Correctness Rate of Transfer Test under Different Conditions.

4.4. Transfer of Fault Diagnosis Algorithm for Rotating Machinery between Different Objects

The fault diagnosis algorithm for different objects is applied by the vibration data of the 0-horsepower drive end bearing and the 0 horsepower lower fan end bearing fault. The source domain is the 0 horsepower lower drive end bearing, and the target domain is the 0 horsepower lower fan end bearing. Different bearing's sizes and other mechanical structures are different. When working in the same working condition, the similarity between the target domain and the source domain is lower than the same bearing working in different working conditions. It can be roughly divided into S3 and S4 according to the size of data in the target domain. The test grouping and transfer methods are the same as the transfer methods under different working conditions.

Figures 15–17 show the training accuracy, training loss, testing accuracy and testing loss of TM1, TM2 and TM3 under four groups of experiments respectively. From the change of training accuracy and training loss, it can be seen that when transfer between different objects, the size of data in the target domain significantly affects the convergence speed and training accuracy of the target domain diagnosis algorithm under each transfer method, which is different from transfer under different working conditions. The larger the size of data in the target domain, the faster the convergence speed of the algorithm, and the higher the final training accuracy. From the change of test accuracy and test loss, it can be seen that when the size of data in the target domain is small, the test loss even shows an upward trend, and only TM2 test loss is relatively stable. With the increase of the target domain data, the performance of the target domain diagnosis algorithm under each transfer method. The larger the data size in the target domain, the better the final fault diagnosis effect of the algorithm.

Figures 18–21 show the comparison of transfer methods in four groups of experiments. It can be seen that TM2 is superior to the other two transfer methods in terms of fitting speed, convergence speed and test accuracy in four target domain data quantities. When the amount of data is small, TM1 is better than TM3. With the increasing amount of data, TM1 has the same performance as TM3.



Figure 15. Results of TM1 between different objects.



Figure 16. Results of TM2 between different objects.



Figure 17. Cont.



Figure 17. Results of TM3 between different objects.



Figure 18. Comparison of three transfer methods under 0.125 data size.



Figure 19. Comparison of three transfer methods under 0.375 data size.



Figure 20. Comparison of three transfer methods under 0.625 data size.



Figure 21. Comparison of three transfer methods under 0.875 data size.

Table 6 summarizes the test accuracy of four groups of experiments. The experiments related to TM1 are performed only once. The experiments related to TM2 and TM3 were repeated five times, and the results shown in the table are the average results of repeated experiments. Comparing the transfer methods, it can be seen that in S3, the performance of TM1, which inherits part of the source network weights, and TM2, which inherits all the source network weights, are higher than TM3, and TM2 has better performance than TM1, that is to say, when the similarity of the source domain of the target domain is low but the size of the target domain data is small, the network structure and weight of the source domain should be transferred as much as possible. Inheriting more source domain information can better guide the target domain algorithm training and improve the fault diagnosis performance of the target domain algorithm. In S4, TM1 and TM3 have comparable performance, but both are lower than TM2. Analyzing the reasons, the difference between the data caused by different bearing differences used in the data is smaller, which made the similarity between the target domain and source domain smaller than supposed, and is closer to S1, so TM2 has better performance, so the performance of TM2 is better. From the perspective of data size, it can be seen that under each transfer method, increasing the data size in the target domain significantly improves the performance of the

algorithm. Therefore, in the case of low similarity between the target domain and the source domain, we should get as much data as possible in the target domain.

Transfer Scenario	Data Ration	Transfer Method 1	Transfer Method 2	Transfer Method 3
S3	0.125	0.5066	0.6283	0.4617
	0.375	0.6581	0.8390	0.5777
S4	0.625	0.8278	0.8450	0.7656
	0.875	0.8791	0.9714	0.8747

Table 6. Summary of Correctness Rate of Transfer Test between Different Objects.

5. Conclusions

This paper develops a deep learning PHM algorithm transfer technology based on transfer learning for the situation of small amount of data in target domain or low similarity between target domain and source domain. By analyzing the structure of the deep learning PHM algorithm, the most valuable content of the algorithm is determined, and the scenario of the transfer is studied. Then, the transfer strategy and its specific implementation scheme which can be directly applied are discussed. In the end, the applicability of the proposed deep learning PHM algorithm transfer technology is verified by the transfer application of the fault diagnosis algorithm under different working conditions (S1, S2) and between different objects (S3, S4).

Network weights and structural designs should be transferred as much as possible under S1, S2, and S3. The test results of S4 show that the boundaries between the scenarios are not very clear. In practical industrial applications, TS1 should be used as much as possible means network structure and weight of source domain should be transferred at the same time, which allows the target domain algorithm to perform better with less data. At the same time, when the similarity between the target domain and the source domain decreases, the target domain data should be collected as much as possible, which has a positive effect on improving the performance of the target domain algorithm.

Author Contributions: Conceptualization, J.M., F.Y., W.Z. and L.T.; methodology, J.M and F.Y.; software, F.Y.; validation, J.M., F.Y. and W.Z.; formal analysis, F.Y.; investigation, J.M. and F.Y.; resources, J.M., W.Z. and L.T.; data curation, F.Y.; writing—original draft preparation, F.Y.; writing—review and editing, J.M.; visualization, F.Y.; supervision, J.M.; project administration, J.M and L.T.; funding acquisition, J.M. and L.T. All authors have read and agreed to the published version of the manuscript.

Funding: This study was supported by the National Natural Science Foundation of China (Grant Nos. 61973011, 61803013, and 51605014), the Fundamental Research Funds for the Central Universities (Grant No. ZG140S1993), National key Laboratory of Science and Technology on Reliability and Environmental Engineering (Grant Nos. 6142004180501, WDZC2019601A304 and 6142004190501), R&D projects in key areas of Guangdong Province (Grant No.2018B010108005), the Research Fund(Grant No. 61400020401), as well as the Capital Science & Technology Leading Talent Program (Grant No. Z191100006119029).

Acknowledgments: Thanks to the open bearing data of Case Western Reserve University.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

- Tsui, K.-L.; Chen, N.; Zhou, Q.; Hai, Y.; Wang, W. Prognostics and Health Management: A Review on Data Driven Approaches. *Math. Probl. Eng.* 2015, 1–17. [CrossRef]
- Lee, J.; Wu, F.; Zhao, W.; Ghaffari, M.; Liao, L.; Siegel, D. Prognostics and health management design for rotary machinery systems—Reviews, methodology and applications. *Mech. Syst. Signal Process.* 2014, 42, 314–334. [CrossRef]
- 3. Sun, W.; Shao, S.; Zhao, R.; Yan, R.; Zhang, X.; Chen, X. A sparse auto-encoder-based deep neural network approach for induction motor faults classification. *Measurement* **2016**, *89*, 171–178. [CrossRef]

- Li, C.; Sánchez, R.-V.; Zurita, G.; Cerrada, M.; Cabrera, D.; Vásquez, R. Multimodal deep support vector classification with homologous features and its application to gearbox fault diagnosis. *Neurocomputing* 2015, 168, 119–127. [CrossRef]
- 5. Tamilselvan, P.; Wang, P. Failure diagnosis using deep belief learning based health state classification. *Reliab. Eng. Syst. Saf.* **2013**, *115*, 124–135. [CrossRef]
- 6. Wang, X.; Li, Y.; Rui, T.; Zhu, H.; Fei, J. Bearing fault diagnosis method based on Hilbert envelope spectrum and deep belief network. *J. Vibroengineering* **2015**, *17*, 1295–1308.
- 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]
- 8. Demetgul, M.; Yildiz, K.; Taskin, S.; Tansel, I.; Yazicioglu, Ö.; Yıldız, K. Fault diagnosis on material handling system using feature selection and data mining techniques. *Meas.* **2014**, *55*, 15–24. [CrossRef]
- 9. Pan, S.J.; Yang, Q. A Survey on Transfer Learning. *IEEE Trans. Knowl. Data Eng.* 2009, 22, 1345–1359. [CrossRef]
- Tan, C.; Sun, F.; Kong, T.; Zhang, W.; Yang, C.; Liu, C. A Survey on Deep Transfer Learning. In Proceedings of the International Conference on Artificial Neural Networks, Rhodes, Greece, 4–7 October 2018; pp. 270–279.
- 11. Wang, M.; Deng, W. Deep visual domain adaptation: A survey. Neurocomputing 2018, 312, 135–153. [CrossRef]
- Saito, K.; Watanabe, K.; Ushiku, Y.; Harada, T. Maximum Classifier Discrepancy for Unsupervised Domain Adaptation. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 19–21 June 2018; pp. 3723–3732.
- 13. Lu, W.; Liang, B.; Cheng, Y.; Meng, D.; Yang, J.; Zhang, T. Deep Model Based Domain Adaptation for Fault Diagnosis. *IEEE Trans. Ind. Electron.* **2017**, *64*, 2296–2305. [CrossRef]
- Xie, J.; Zhang, L.; Duan, L.; Wang, J. On cross-domain feature fusion in gearbox fault diagnosis under various operating conditions based on Transfer Component Analysis. In Proceedings of the 2016 IEEE International Conference on Prognostics and Health Management (ICPHM), Ottawa, ON, Canada, 20–22 June 2016; pp. 1–6.
- 15. Wen, L.; Gao, L.; Li, X. A New Deep Transfer Learning Based on Sparse Auto-Encoder for Fault Diagnosis. *IEEE Trans. Syst. Man, Cybern. Syst.* **2017**, *49*, 136–144. [CrossRef]
- Han, T.; Liu, C.; Yang, W.; Jiang, D. Deep transfer network with joint distribution adaptation: A new intelligent fault diagnosis framework for industry application. *ISA Trans.* 2020, 97, 269–281. [CrossRef] [PubMed]
- 17. Yang, B.; Lei, Y.; Jia, F.; Xing, S. An intelligent fault diagnosis approach based on transfer learning from laboratory bearings to locomotive bearings. *Mech. Syst. Signal Process.* **2019**, 122, 692–706. [CrossRef]



© 2020 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/).