Article:
Liu, Z.-H., Lu, B.-L., Wei, H.-L. et al. (2021) A stacked auto-encoder based partial adversarial domain adaptation model for intelligent fault diagnosis of rotating machines. IEEE Transactions on Industrial Informatics, 17 (10). pp. 6798-6809. ISSN: 1551-3203

https://doi.org/10.1109/tii.2020.3045002

Manuscript accepted (8 December 2020) for publication by *IEEE Transactions on Industrial Informatics*. DOI: 10.1109/TII.2020.3045002
IEEE-TII Early Access link: https://ieeexplore.ieee.org/document/9294067

A Stacked Auto-Encoder Based Partial Adversarial Domain Adaptation Model for Intelligent Fault Diagnosis of Rotating Machines

Zhao-Hua Liu, *Member, IEEE*, Bi-Liang Lu, Hua-Liang Wei, Lei Chen, Xiao-Hua Li, and Chang-Tong Wang

***Abstract*—Fault diagnosis plays an indispensable role in prognostics and health management of rotating machines. In recent years, intelligent fault diagnosis methods based on domain adaptation technology have attracted the attention of researchers. However, a more extensive application scenario of fault diagnosis - partial domain adaptation (PDA), has not been well resolved. In this paper, for the first time a novel stacked auto-encoder based partial adversarial domain adaptation (SPADA) model is proposed to solve the fault diagnosis problem in PDA situations. Two deep stack auto-encoders are first designed to extract representative features from the training data (source domain) and test data (target domain), respectively. Then, a weighted classifier based on Softmax is used to weight the features from the source and target domains. Meanwhile, another domain discriminator and label predictor using the Softmax classifier are adopted to simultaneously implement domain adaptation and fault diagnosis. Comprehensive analysis is performed on real data to test the performance of the SPADA model and detailed comparisons are provided; the extensive experimental results show that the diagnosis performance of SPADA outperforms the existing deep learning and domain adaptation methods in dealing with the PDA problem.**

***Index Terms*—Deep learning, domain adaptation, fault diagnosis, machine learning, partial adversarial domain adaptation, rolling bearing, rotating machines, softmax classifier, stack auto-encoder (SAE).**

I. Introduction[1]

With the development of Industry 4.0 and smart manufacturing, rotating machines are becoming increasingly complex and precise. Traditional fault diagnosis methods, such as model-based analysis and signal processing may no longer be suitable for future applications [1], as the fault diagnosis system should be designed by exploiting big data with intelligent learning approaches in Industry 4.0 [2]. In recent years, the advance of computing power and deep learning techniques makes it possible to adopt intelligent fault diagnosis methods using big data [3], [4]. Meanwhile, the core idea of intelligent fault diagnosis based on deep learning is to use a variety of data sources (e.g. vibration data, temperature, pressure and so on) collected from different devices by sensors to train a deep learning model; then, the trained model is used for rotating machines fault diagnosis. There are many rotating machines whose diagnosis and health monitoring benefit from this, including electric locomotive, wind turbines, high-speed railway, and so on.

A common challenge encountered by any data-based fault diagnosis method is that it usually requires that the main statistical properties or features of the signals of interest should be maintained during the entire monitoring process. That is to say, the training data (source domain) and the test data (target domain) need to obey the same distribution [5]. In other words, the rotating machines for training data and test data must be collected under the same working conditions. Obviously, it is impossible to strictly satisfy such requirements in practical fault diagnosis applications. Given that the cost of collecting and re-collecting fault data is very high, it is desirable to use existing fault data to train models and then use these models to diagnose new rotating machines. However, models trained by direct using these existing data may lack generalization performance and thus result in low detection and classification accuracy. Therefore, the distribution difference between source domain (SD) and test set target domain (TD) would be a big challenge in most engineering practices, and such an issue is defined as domain shift problem in the transfer learning area [6].

Domain adaptation (DA), as a commonly used strategy in deep learning and transfer learning, can effectively solve the learning problem of inconsistent probability distribution of SD and TD [7], [8]. Its main purpose is to improve the performance of models trained using data with different but related TD distributions from SD. Usually, the domain adaptation methods use a predefined domain shift distance loss, and then reduce the distance loss between the SD and the TD to achieve domain matching. Many predefined distance loss methods have been proposed in recent years, including maximum mean discrepancy (MMD), Kullback-Leibler (KL) divergence, Wasserstein distance, etc. For instance, Lu *et al.* [9] proposed an auto-encoder (AE) network-based deep-domain adaptation model for rolling bearings fault diagnosis, in which the MMD is embedded in the loss function of the AE network. Qian *et al.* [10] proposed a novel dataset distribution discrepancy measuring algorithm based on high-order KL (HKL)

[1] Manuscript received August 8, 2020; revised October13, 2020, November 23, 2020, and accepted December 8, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61972443, Grant 61503134, National Key Research and Development Project under Grant 2019YFE0105300, in part by the Hunan Provincial Hu-Xiang Young Talents Project of China under Grant 2018RS3095, in part by the Hunan Provincial Natural Science Foundation of China under Grant 2018JJ2134 and Grant 2020JJ5199, and in part by the Scientific Research Fund of Hunan Provincial Education Department under Grant 18C0296.

Z.-H. Liu, B.-L. Lu, L. Chen, X.-H. Li, and C.-T. Wang are with the School of Information and Electrical Engineering, Hunan University of Science and Technology, Xiangtan, 411201, China (e-mail: zhaohualiu2009@hotmail.com, 1197393632@qq.com, chenlei@hnust.edu.cn, lixiaohua_0227@163.com, ctwang@mail.hnust.edu.cn).

H.-L. Wei is with the Department of Automatic Control and Systems Engineering, the University of Sheffield, Sheffield S1 3JD, U.K (e-mail: w.hualiang@sheffield.ac.uk).

Manuscript accepted (8 December 2020) for publication by *IEEE Transactions on Industrial Informatics*. DOI: 10.1109/TII.2020.3045002
IEEE-TII Early Access link: https://ieeexplore.ieee.org/document/9294067

divergence. Tong *et al.* [11] proposed a novel domain adaptation model by using feature transfer learning (DAFTL) to solve the performance degradation issue. However, these domain adaptation methods have clear restrictions on the SD and TD: the number of failure samples and types of failures in the SD and TD need to be consistent. In practice, there are some challenges for using these methods, for example, the data set may have few or no target data in the training phase, and the SD and TD possibly have different types of faults. For such a case, traditional domain adaptation techniques may fail, since they all need to align the amount and type of data in the SD and TD. In conclusion, there are mainly two defects for these domain adaptation methods for fault diagnosis: 1) Inconsistent data sizes in the SD and TD can cause distribution mismatching problem. 2) Inconsistent types of samples in the SD and TD can cause partial domain adaptation (PDA) problem.

Recently, adversarial domain adaptation (ADA) [12] has attracted considerable attention, because it can achieve distribution matching when the data volume of the TD is scarce. The novel thought behind the ADA is to train two neural networks: a discriminator network trying to distinguish the transformed source domain from the target domain, and a generator network trying to make the SD as close to TD as possible to confuse the discriminator network. It mainly uses the generational adversarial strategy of the generative adversarial network (GAN). By distinguishing the distribution of SD and TD, the transformed SD is similar to the TD distribution, so as to achieve the domain adaptation [13]-[16]. Most recently, many interesting ADA approaches based on GAN have been proposed. For example, Xie *et al.* [17] used a cycle-consistent GAN to solve the rotation machinery fault diagnosis problem. Liu *et al.* [18] proposed a small-sample wind turbine fault detection method using the GAN. Liang *et al.* [19] proposed an intelligent fault diagnosis method via semi-supervised GAN and wavelet transform. Nevertheless, ADA solves the domain adaptation problem under the condition that the TD samples are lacking, but it still has not achieved good results for the domain adaptation learning problem for the PDA scenario which is briefly described below.

PDA assumes that the SD and TD have different category spaces as shown in Fig. 1; this phenomenon is very common in the actual working conditions of rotating machines [20], [21]. For example, the vibration data collected from the rotating machines health management system is likely to contain both bearing and gearbox data. If these data are directly used for the rolling bearing fault diagnosis of new rotating machines, the gearbox data might cause negative transfer. Consequently, the PDA problem is better in line with the actual rotating machines health detection scenario, where the sample type of TD to be diagnosed is the subspace of the SD. Therefore, it is challenging and significant to deal with this PDA problem in intelligent fault diagnosis [22].

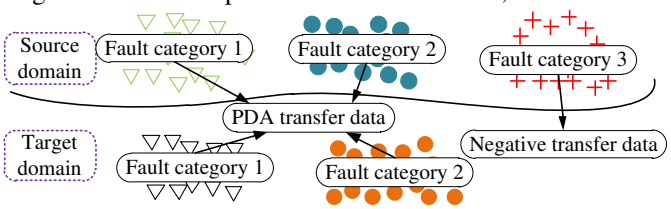


Fig. 1. A diagram of partial transfer learning.

In this paper, a novel stacked auto-encoder based partial adversarial domain adaptation (named SPADA) model is proposed for intelligent fault diagnosis of rotating machines in the PDA scenario. The proposed method uses weighted domain classifiers to distinguish negative transfer samples in SD, and the adversarial training of discriminators minimizes the approximate domain discrepancy distance. At the same time, the SPADA uses an unsupervised learning method and can be trained through an end-to-end network. Such a SPADA model has several advantages for rotating machines fault diagnosis. Firstly, it can effectively solve the data inconsistency issue where SD and TD have inconsistent distribution in the PDA scenario. Second, it can easily be implemented for real applications. To the best of our knowledge, this is a novel work that addresses rotating machine fault diagnosis problem through adversarial networks in the PDA scenario. The main contributions are summarized as follows:

1) A novel stacked auto-encoder based partial adversarial domain adaptation model (SPADA) is proposed, which constructs a weighted domain discriminator to weight the sample space of SD and aligns it with the TD. The domain adaptation can be implemented using the ADA method, meanwhile a multilayered network can also be used to learn rich knowledge of the source domain to promote domain adaptation.

2) In order to effectively solve the problem of intelligent fault diagnosis of rotating instruments in the PDA scenario, a novel deep adversarial domain adaptation algorithm based on SPADA model is designed. Results from case studies show that the proposed method can significantly improve the accuracy of fault diagnosis in the PDA scenario.

3) Comprehensive experiments are performed based on two benchmark datasets and eighteen partial domain adaptation learning scenarios. Furthermore, the effects of the number of hidden layers in the stack auto-encoder (SAE) network, the number of neurons in each hidden layer, and the hyper-parameters of the SPADA on the model performance are analyzed. The experimental results show that the SPADA can produce excellent classification accuracy and has strong domain adaptation ability.

The remaining parts of this paper are summarized as follows. The proposed SPADA model is presented in Section II. In Section III, we provide a novel partial adversarial domain adaptation algorithm for fault diagnosis of rotating machines, as well as its optimal solution. Experimental results on different domain adaptation situations are displayed in Section IV. At last, a brief summary is given in Section V.

II. The Proposed Partial Adversarial Domain Adaptation Model

Fault diagnosis methods based on deep learning can usually achieve good results under the assumption - that SD and TD have the same distribution [23]. Unfortunately, this is difficult to meet in practice. Traditional DA approaches can be used to solve the domain shift problem in SD and TD. However, it cannot achieve good DA performance in the PDA scenarios since the TD is a subspace of SD. In recent years, ADA has

Manuscript accepted (8 December 2020) for publication by *IEEE Transactions on Industrial Informatics*. DOI: 10.1109/TII.2020.3045002
IEEE-TII Early Access link: https://ieeexplore.ieee.org/document/9294067

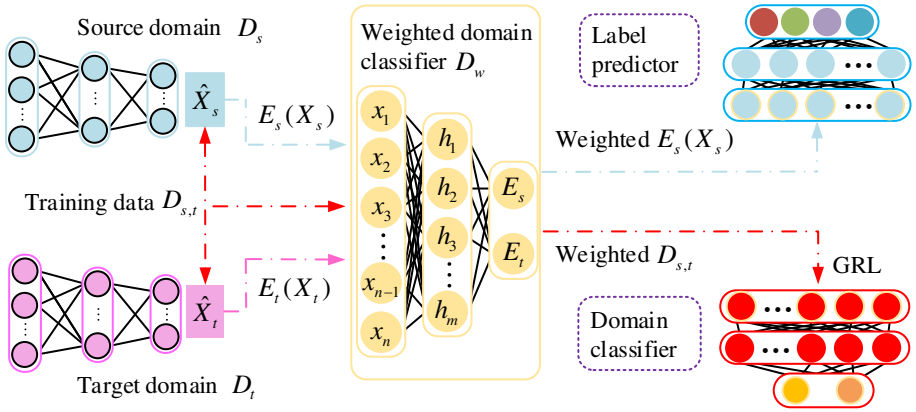


Fig. 2. The proposed partial adversarial domain adaptation model.

attracted increasing interest due to its excellent generative performance. The use of ADA for fault diagnosis can help solve the following difficult problem: SD and TD have inconsistent distributions and contain imbalanced failure samples. But it still has the same deficiencies as other DA methods.

This section introduces a SPADA model to resolve the domain shift problem in the PDA situation. The architecture of SPADA is shown in Fig. 2, where $\hat{X}_s$ and $\hat{X}_t$ are the features of the SD and TD. The previous SAE structures are their respective feature extractors. The yellow part is the first domain classifier used to obtain the importance weight of the source sample. The red part (at the right bottom corner) is the second domain classifier, which uses weighted source domain samples and target samples for minimax games. GRL represents the gradient reversal layer, which acts as an identity conversion in the forward propagation and changes the sign of the gradient in the backward propagation. The four circles of different colors on the label classifier (the panel at the right top corner) represent the four categories.

The main difference between SPADA and ADA is that the former contains a domain classifier for weighting samples in the SD. In this way, the data from the sample space in different domains will be filtered by the weighted classifier. Basically, the SPADA architecture consists of several SAEs and three Softmax classifiers (named discriminator). The core idea of SAE is to use the SD and TD samples to generate its representative features. Thus, SAE is often referred to as a generator. In the present study, two different generators are used to extract data from the SD and TD. The data from the source domain is only weighted by the weighted domain classifier D_w; the other two classifiers (domain classifier D_d and label predictor D_l) implement domain adaptation and fault type prediction, respectively. Actually, the training process of the entire model is a player game between the generator and the discriminator. The generator (denoted by *G*) can generate representative features to confuse the discriminator (denoted by *D*), so that the *D* cannot distinguish whether the representative feature belongs to the SD or the TD, and the other player *D* is struggling to distinguish the field to which the representative feature belongs. Among them, the weighted classifier does not participate in the iterative training process; it only needs to perform weighted processing on the SD data.

Similar to the original GAN, the ADA nets-based domain adaptation model uses the following minimax loss:

$$\min_{f_s,f_t}\max_{D}\mathcal{L}(D,f_s,f_t)=E_{x\sim p_s(x)}[\log D(f_s(x))]+E_{x\sim p_t(x)}[\log(1-D(f_t(x)))] \tag{1}$$

where f_s and f_t are feature extractors for SD data and TD data, and D is a domain classifier. Maximizing the loss of D will make the SD distribution closer to the TD, while minimizing the loss regarding to the f parameters will minimize the divergence in features distribution. In this paper, we introduce a weighted domain classifier D_w to weight SD data only. Given the representative feature of data $f_{s,t}(x)$, and the similarity between SD samples and the TD can be defined by the formula as follows:

$$D_w(f_{s,t}(x))=p(y=1\big|f_{s,t}(x))=\sigma(\alpha(f_{s,t}(x))) \tag{2}$$

where σ is the logistic sigmoid function, and y is the label of the input feature. Suppose the probability that a classified feature is labeled as SD is 1, then $D_w^*(f_{s,t}(x))\approx 1$, meaning that the probability of the features come from the outlier class in the SD is close to 1, and the shared class between both domains is close to 0. These features should be given a smaller weighting value to reduce the sharing of domain shift. Hence, the features weighting function should be inversely proportional to $D_w^*(f_{s,t}(x))$, therefore the entire weighting process can be written as follows:

$$\tilde{\mu}(f_{s,t}(x))=1-D_w^*(f_{s,t}(x)) \tag{3}$$

Given $f_s(x)$ (corresponding to real samples in GAN), for any $f_t(x)$ (corresponding to generated samples in GAN), the optimum D is obtained at:

$$D_w^*(f_{s,t}(x))=\frac{p_s(f_{s,t}(x))}{p_s(f_{s,t}(x))+p_t(f_{s,t}(x))} \tag{4}$$

where $f_s(x)$ is representative features after feature extraction networks. Similar to [24], the proof of (4) is given as follows:

Proof. For any $f_s(x)$ and $f_t(x)$, the domain classifier D is to maximize (1):

Manuscript accepted (8 December 2020) for publication by *IEEE Transactions on Industrial Informatics*. DOI: 10.1109/TII.2020.3045002
IEEE-TII Early Access link: https://ieeexplore.ieee.org/document/9294067

$$\begin{aligned}\max_{D}\mathcal{L}(D,f_s,f_t)=&\int_x p_s(x)\log D(f_s(x))\\&+p_t(x)\log(1-D(f_t(x)))dx\\=&\int_{f_{s,t}(x)} p_s(f_{s,t}(x))\log D(f_{s,t}(x))\\&+p_t(f_{s,t}(x))\log(1-D(f_{s,t}(x)))df_{s,t}(x)\end{aligned}\tag{5}$$

For any $(a,b)\in\mathbb{R}^2\setminus\{0,0\}$, the function $y\to a\log(y)+b\log(1-y)$ achieves its maximum in [0, 1] at $\frac{a}{a+b}$.

Thus, equation (3) can be rewritten as:

$$\tilde{\mu}(f_{s,t}(x))=1-D_w^*(f_{s,t}(x))=\frac{1}{\frac{p_s(f_{s,t}(x))}{p_t(f_{s,t}(x))}+1}\tag{6}$$

It can be seen that if the value of $D_w^*(f_{s,t}(x))$ becomes large, $\tilde{\mu}(f_{s,t}(x))$ will become small, so the ratio $\frac{p_s(f_{s,t}(x))}{p_t(f_{s,t}(x))}$ will be large. Therefore, the weight of the SD samples from the anomalous class will be less than the weight of the shared class. After adding the weighted domain classifier, the objective function of the original ADA can be rewritten as:

$$\begin{aligned}\min_{f_t}\max_{D_w}\mathcal{L}(D_w,f_s,f_t)=&E_{x\sim p_s(x)}[\tilde{\mu}(f_s(x))\log D_w(f_s(x))]\\&+E_{x\sim p_t(x)}[\log(1-D_w(f_t(x)))]\end{aligned}\tag{7}$$

where $\tilde{\mu}(f_s(x))$ is a weighing function on the logarithm of $D_w^*(f_s(x))$. The remaining domain classifier D_d and the label predictor D_l are defined as follows:

$$\begin{aligned}E(\theta_f,\theta_y,\theta_d)=&\sum_{X_i\in\mathcal{D}_s}L_y(G_y(G_f(X_i;\theta_f);\theta_y),y_i)-\\&\lambda\sum_{X_i\in\mathcal{D}_s\cup\mathcal{D}_t}L_d(G_d(G_f(X_i;\theta_f);\theta_d),y_i)\end{aligned}\tag{8}$$

where X_i is the weighted source domain features, L_y is the loss of the label predictor D_l, L_d is the loss of the domain discriminator D_d, y_i is the label for X_i, and λ is a trade-off parameter. During the training progress, the parameters $\hat{\theta}_f$, $\hat{\theta}_y$, and $\hat{\theta}_d$ are updated iteratively using stochastic gradient descent (SGD) algorithm based on the weighted source domain data and target domain data. The operation of the gradient reversion layer (GRL) ensures that the generator loss is maximized while the discriminator loss is minimized, and this process can be written as:

$$(\hat{\theta}_f,\hat{\theta}_y)=\arg\min_{\theta_f,\theta_y}E(\theta_f,\theta_y,\hat{\theta}_d)\tag{9}$$

$$\hat{\theta}_d=\arg\max_{\theta_d}E(\hat{\theta}_f,\hat{\theta}_y,\theta_d)\tag{10}$$

where E denotes the mathematical expectation.

Hence, the overall objective of the weighted adversarial nets-based method is:

$$\begin{aligned}\min_{f_t}\max_{D_w}\mathcal{L}(D_w,f_s,f_t)=&E_{x\sim p_s(x)}[\tilde{\mu}(f_s(x))\log D_w(f_s(x))],\\&+E_{x\sim p_t(x)}[\log(1-D_w(f_t(x)))]\end{aligned}$$

$$(\hat{\theta}_f,\hat{\theta}_y)=\arg\min_{\theta_f,\theta_y}E(\theta_f,\theta_y,\hat{\theta}_d),\tag{11}$$

$$\hat{\theta}_d=\arg\max_{\theta_d}E(\hat{\theta}_f,\hat{\theta}_y,\theta_d).$$

The following rules are used to update the parameters throughout the training process:

$$\theta_f\leftarrow\theta_f-\mu\left(\frac{\partial L_y^i}{\partial\theta_f}-\lambda\frac{\partial L_d^i}{\partial\theta_f}\right),\tag{12}$$

$$\theta_y\leftarrow\theta_y-\mu\frac{\partial L_y^i}{\partial\theta_y},\tag{13}$$

$$\theta_d\leftarrow\theta_d-\mu\frac{\partial L_d^i}{\partial\theta_d}.\tag{14}$$

where μ is the learning rate, $\frac{\partial L_y^i}{\partial\theta_y}$, $\frac{\partial L_d^i}{\partial\theta_f}$, and $\frac{\partial L_d^i}{\partial\theta_d}$ are the partial derivatives of the loss function to the $\hat{\theta}_f$, $\hat{\theta}_y$, $\hat{\theta}_d$ parameter, and $-\lambda$ means inverting the gradient. In fact, the gradient reversion is performed between the generator SAE and the domain discriminator connection layer, and is defined as the GRL.

III. The Partial Adversarial Domain Adaptation Method for Rotating Machines

The training data (denoted by C_s) and test data (denoted by C_t) collected by rotating machines usually contain different types of faults, meaning that $C_s\neq C_t$. As mentioned earlier, traditional domain adaptation methods assume that the types of faults match between the SD and TD, that is, $C_s=C_t$. However, under some complex working conditions, the types of faults are often inconsistent in the SD and TD, and a common case would be that $C_t\subset C_s$. In order to improve the accuracy of fault diagnosis and the generalization ability of the model, a new algorithm for fault diagnosis of rotating machines is proposed, based on the partial adversarial domain adaptation network. The algorithm structure is shown in Fig. 3. There are several aspects that need to be explained. At the beginning, the SD and TD enter the feature extractor SAE to obtain their respective representative features. In this feature space, the features of two domains are mixed together. The feature space also contains gearbox data, which will produce negative transfer. Then, the SD features are processed by the weighted domain classifier, and gearbox data can be separated from the feature space. Finally, the adversarial training of the another domain classifier makes SD and TD match the distribution.

A. Feature Generator Using Stacked Auto-Encoder Network

Deep learning has seen a dramatic revival in past 7 years also, and AE, as a special type of unsupervised neural networks, is very effective for reducing dimensionality in a nonlinear space. AE is significantly better than other linear

Manuscript accepted (8 December 2020) for publication by *IEEE Transactions on Industrial Informatics*. DOI: 10.1109/TII.2020.3045002
IEEE-TII Early Access link: https://ieeexplore.ieee.org/document/9294067

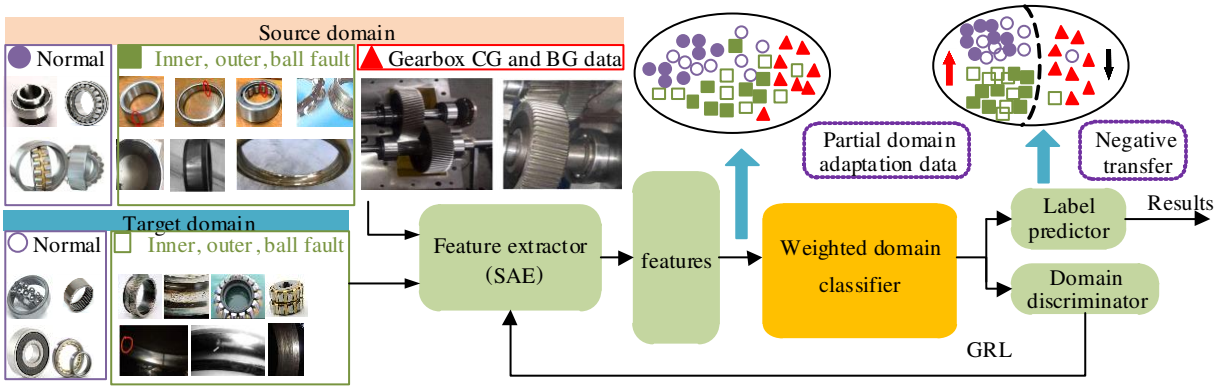


Fig. 3. The proposed partial adversarial domain adaptation algorithm. CG and BG denote the chipped tooth gear and broken tooth gear, respectively.

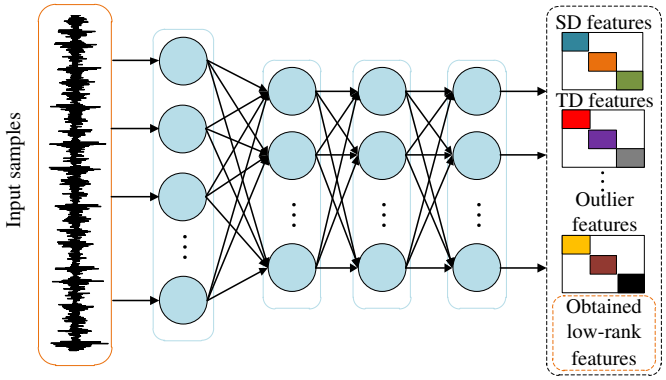


Fig. 4. SAE network.

transformation dimensionality reduction methods such as principal component analysis (PCA), linear discriminant analysis (LDA), etc. Because the former is more flexible and allows for taking nonlinear and linear transformations over the data, while the latter only uses linear transformations. SAE is a combination of multilayer AE network; comparing with a typical AE network, it can extract deeper information hidden under the original data to form more abstract and advanced representative features. The main structure of SAE is shown in Fig. 4.

Given a set of training samples without category labels $x=\left\{x^{(1)},x^{(2)},x^{(3)},.....,x^{(n)}\right\}$, where $x^{(i)}\in R^n$, the main operation performed by the AE network is as follows: the target value is equal to the input value, meaning that $y^{(i)}=x^{(i)}$. The coding part of the AE network can be defined as:

$$\begin{aligned} h &= F(\omega^{(c)}x+b^{(c)}),\\ F(x) &= 1/1+\exp(x). \end{aligned} \tag{15}$$

where $\omega^{(c)}$ and $b^{(c)}$ are the weight matrix of size $n\times m$ and the deviation vector of size m, respectively. $F(x)$ is the activation function such as sigmoid. The decoder layer reconstructs the representative features into vectors as follows:

$$y=F(\omega^{(d)}h+b^{(d)}) \tag{16}$$

where $\omega^{(d)}$ is the weight matrix of size $m\times n$ and the deviation vector $b^{(d)}$ which can the same as $b^{(c)}$. Thus, the entire loss function can be written as:

$$J(\omega,b)=\frac{1}{n}\sum_{i=1}^{n}\left(\frac{1}{2}\left\|h_{\omega,b}(x^{(i)})-y^{(i)}\right\|^2\right) \tag{17}$$

where n is the number of training samples, $\|\ \|^2$ is the L2 norm.

B. Domain Discriminator and Classifier Based on Softmax

Softmax is a popular and useful function in deep learning, especially in multiclassification scenarios. It maps the inputs to real numbers between 0 and 1, and such normalization guarantees that the sum is 1. So the sum of the probabilities of multiple classifications is also exactly equal to 1. Such a feature is in line with the requirements of the weighted domain classifier, and due to its simple structure, it is very suitable as a discriminator for SPADA model. Specially, given input data $\mathcal{D}_s=\{(x^{(1)},y^{(1)}),(x^{(2)},y^{(2)}),\cdots(x^{(n_s)},y^{(n_s)})\}$, with multiple labels $y^{(i)}=\{1,2,\cdots k\}$, $i=1,2,\cdots n_s$, the way that Softmax achieves classification can be expressed as:

$$\begin{aligned} h_\theta(x^{(i)}) &= \left[p(y^{(i)}=1\left|x^{(i)};\theta\right.),\cdots,p(y^{(i)}=k\left|x^{(i)};\theta\right.)\right]'\\ &= \frac{1}{\sum_{l=1}^{k}e^{\theta_l^T x^{(i)}}}\begin{bmatrix} e^{\theta_1^T x^{(i)}}\\ e^{\theta_2^T x^{(i)}}\\ \cdots\\ e^{\theta_k^T x^{(i)}}\end{bmatrix} \end{aligned} \tag{18}$$

where $\theta=\{\theta_1,\theta_2,\cdots\theta_k\}$ represents the parameters of Softmax, which is obtained by optimizing the objective function of each sample, θ^T is the transpose of θ, $1/\sum_{l=1}^{k}e^{\theta_l^T x^{(i)}}$ is a hypothetical function.

C. Training Process and Optimization Strategy

The training process of SPADA is divided into two stages. The first stage is the data transmission from low to high level, that is, the forward propagation stage. During the forward propagation process, the input data of the rotating equipment is encoded by the multilayer SAE network, and the feature vector is defined. The feature vector is passed into the weighted domain classifier to obtain the weighted result of the source domain samples. Then, the weighted source domain sample data is used to implement domain adaptation, and the label predictor implements data fault diagnosis. The second stage is to carry out error training from the high to the low level, when the result of forward propagation does not match the expectation; this is referred to as the back propagation phase.

Manuscript accepted (8 December 2020) for publication by *IEEE Transactions on Industrial Informatics*. DOI: 10.1109/TII.2020.3045002
IEEE-TII Early Access link: https://ieeexplore.ieee.org/document/9294067

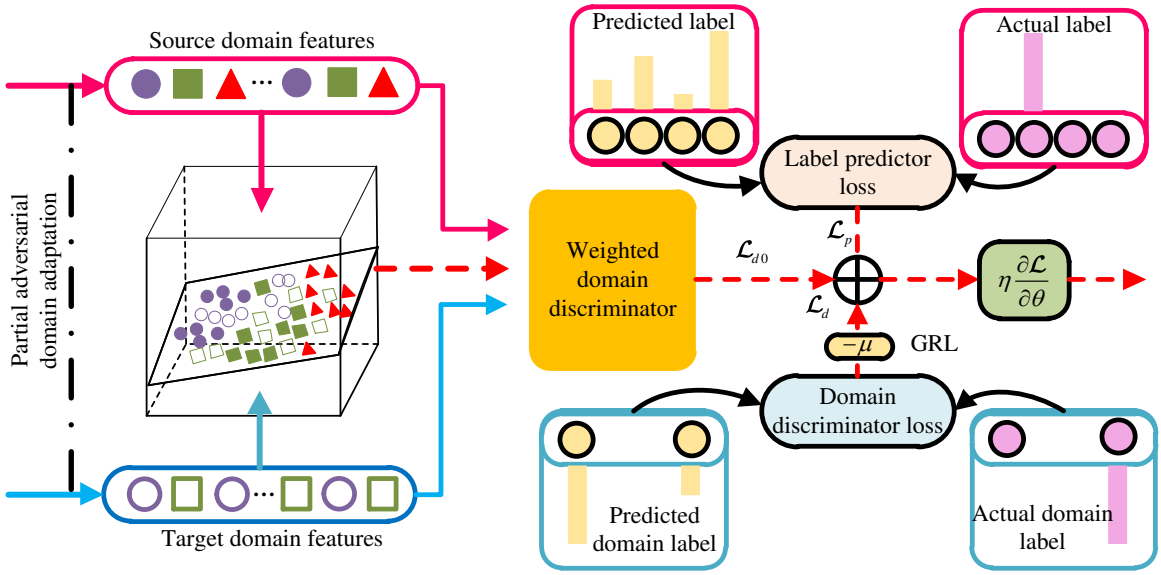


Fig. 5. The training process of the partial domain adaptation.

The training process is shown in Fig. 5. In order to facilitate the reproduction of the algorithm, its pseudo code is provided in Algorithm 1.

Algorithm 1: Partial adversarial domain adaptation algorithm

***Input*:** Labeled source domain data $x_s = \{(x^{(1)}, y^{(1)}),...,(x^{(n_s)}, y^{(n_s)})\}$ and unlabeled target domain data $x_t = \{x^{(1)}, x^{(2)},...,x^{(n_t)}\}$ (assume that $x_t \subset x_s$).

***Output*:** Target domain data label $\{y_i\}_{i=1}^{n^t}$.

1: Randomly initialize SAE network parameters θ :
2: **for** epoch $i = 1$ **to** N **do**
3: **for** k **in**{SAE, $\mathcal{L}_{dw}$, $\mathcal{L}_d$, $\mathcal{L}_p$ } **do**
4: Obtain SAE features: $\{F_i\}_{i=1}^{n^s+n^t} = trained\_SAE\left(X_S \cup X_t\right)$.
5: **end for**
6: The SGD is used to optimize label predictors G_y based on the labeled source domain samples only $\{F_i\}_{i=1}^{n^s}$.
7: The weighted domain classifier D_w is used for sample weighting on SAE feature samples $\{F_i\}_{i=1}^{n^s+n^t}$.
8: The weighted sample features $\omega \times \{F_i\}_{i=1}^{n^s+n^t}$ are input to another domain classifier.
9: **for** k **in**{ $\mathcal{L}_d$, SAE} **do**
10: Update SAE network parameters θ and domain classifier parameters θ_d according to (9)-(11).
11: **end for**
12: **end for**
13: **return** unlabeled $\{F_i\}_{i=1}^{n^t}$ corresponding labels $\{y_i\}_{i=1}^{n^t}$.

IV. Experimental Test

A. Experimental Data Description

To demonstrate the performance of the proposed method for fault diagnosis of rotating machines, two publically available and commonly used datasets are used to carry out numerical experiments. The two datasets are briefly described below.

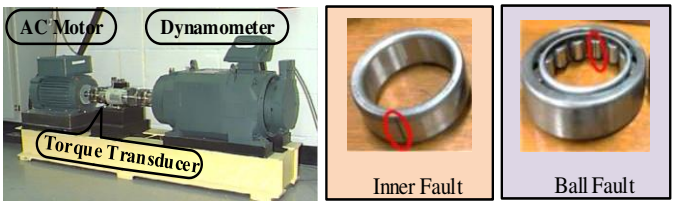


Fig. 6. The bearing test rig and fault types [25].

Fig. 7. The Gearbox test rig and fault types [26].

Dataset I: CWRU. The dataset is provided by the Case Western Reserve University (CWRU) Bearing Data Center [25], and the experimental device is shown in Fig. 6, where the accelerometer is installed at the end of the bearing to obtain normal and fault data, and the dataset includes: 1) normal, 2) outer race fault (OF), 3) inner race fault (IF), and 4) ball fault (BF). Of course, the above three kinds of fault data have four fault diameters (0.007, 0.014, 0.021, and 0.028). In order to facilitate the classification of PDA tasks, the bearing fault diameter is used as a marker to distinguish PDA tasks and assign it a fault size. For example, the code '0.007 fault size' means 9 PDA tasks composed of 0.007 fault diameter bearing fault data and gearbox data.

Dataset II: Gearbox. The focus of this rotating machines dataset contains the accelerometer data and information about the bearing geometry to perform fault detection and size estimation on the universal gearbox [26]. The experimental device is shown in Fig. 7. The gearbox mainly includes three parts: 1) gear, 2) bearing, and 3) shaft. The three types of data are normal gear, chipped tooth gear (CG), and broken tooth gear (BG) that are included in the tachometer information, and the fault data is collected under high load and low load

Manuscript accepted (8 December 2020) for publication by *IEEE Transactions on Industrial Informatics*. DOI: 10.1109/TII.2020.3045002
IEEE-TII Early Access link: https://ieeexplore.ieee.org/document/9294067

conditions. In addition, there are five different settings (30Hz, 35Hz, 40Hz, 45Hz, and 50Hz) of the shaft speed. The fault labels of gearbox data under different shaft speeds and loads are different from that of CWRU data. In this study, the gearbox data are regarded as the anomalous classes for the target domain when constructing the PDA task, meaning that these data are the unshared classes between the source and target domains. It is worth highlighting that for the proposed SPADA model in PDA scenarios, the CWRU data under different fault sizes and loads are considered as the shared classes of the source and target domains. The CG and BG data of Gearbox collected under a single one working condition (i.e. 45Hz shaft speed and low load called 45L) are considered as the anomalous classes to expand the label space of the source domain and cause negative transfer. Finally, the combination of CWRU and Gearbox data becomes the source and target domain in the experiment. From normal, inner fault, outer fault, ball fault, CG, and BG, the order is marked as: 1→2→3→4→5→6. In this study, the fault size (0.007 and 0.014) and four different loads (0, 1, 2, and 3hp) are chosen to simulate the scenario for PDA, and it constitutes a total of eighteen PDA scenarios as shown in Table I.

TABLE I
THE PDA TASK DESCRIPTION

0.007 Fault size			0.014 Fault size		
PDA task	SD- >TD	Source classes	PDA task	SD- >TD	Source classes
D1	1hp- >3hp	1, 2, 3, 4, 5, 6	T1	1hp- >3hp	1, 2, 3, 4, 5, 6
D2	1hp- >3hp	1, 2, 3, 4, 5	T2	1hp- >3hp	1, 2, 3, 4, 5
D3	1hp- >3hp	1, 2, 3, 4, 6	T3	1hp- >3hp	1, 2, 3, 4, 6
D4	0hp- >2hp	1, 2, 3, 4, 5, 6	T4	0hp- >2hp	1, 2, 3, 4, 5, 6
D5	0hp- >2hp	1, 2, 3, 4, 5	T5	0hp- >2hp	1, 2, 3, 4, 5
D6	0hp- >2hp	1, 2, 3, 4, 6	T6	0hp- >2hp	1, 2, 3, 4, 6
D7	1hp- >2hp	1, 2, 3, 4, 5, 6	T7	1hp- >2hp	1, 2, 3, 4, 5, 6
D8	1hp- >2hp	1, 2, 3, 4, 5	T8	1hp- >2hp	1, 2, 3, 4, 5
D9	1hp- >2hp	1, 2, 3, 4, 6	T9	1hp- >2hp	1, 2, 3, 4, 6

Source domain: Normal and fault data under 0hp motor load in CWRU. In this paper, the fault diameter is selected to be 0.007 and 0.014. The CG and BG data of Gearbox collected under 45L working condition are selected as the anomalous classes. Finally, $\Omega_S = \left\{normal^0, IN^0_{0.007,0.014}, OU^0_{0.007,0.014}, BA^0_{0.007,0.014}, CG, BG\right\}$.

Target domain: The target domain contains normal and defective data for the 3hp motor load, but the difference is that there is normal label data only $\Omega_T = \left\{normal^3, IN^3_{0.007,0.014}, OU^3_{0.007,0.014}, BA^3_{0.007,0.014}\right\}$.

Task: Get the labels of unlabeled data in the target domain $y^{(i)} = \{1, 2, \cdots k\}$.

B. Implementation Details

In this paper, two rotating machinery datasets are considered to test the proposed PDA learning schemes, where multiple partial domain adaptation tasks are performed. These tasks cover different categories randomly selected unsupervised samples on the target domain. The fault diagnosis tasks under different PDA learning schemes are given in the Table I. For comparison purposes, several state-of-the-art DA learning methods are used in the experiments, including supervised learning method - support vector machines (SVM) [27], joint domain adaptation - joint distribution adaptation (JDA) [28], adversarial domain adaptation - deep adversarial domain adaptation (DADA) [29]. Furthermore, two methods based on PDA learning are also used for verification, namely, the selective adversarial networks (SAN) [30] and multi-adversarial domain adaptation (MADA) [31]. More specifically, the experimental settings and details of these methods are summarized as follows:

1) For SVM, Gaussian kernel function was used and the experiments were carried out using Matlab 2016. For all the adjustable parameters, we used the default values suggested by the software.

2) For JDA, the AE network, combined with JDA, was used to solve the domain shift problem, where the number of hidden layers of the AE network is 2, and the learning rate is 1×10^{-3}.

3) For DADA, the SAE network was used as a generator, and Softmax was used as a discriminator for adversarial training. The SAE network has a double-layer structure and the learning rate is 5×10^{-3}.

4) For SAN and MADA, the framework is composed of a convolutional neural network and multiple weighted domain classifiers. Weighted for multiple classes separately to achieve partial domain adaptation, more details can be found in [30] and [31], and they run on PyTorch and Caffe respectively.

5) For SPADA, the details of the parameters are shown in the Table II. In addition, it should be noted that under different training data, the optimization of parameters needs to be adjusted.

TABLE II
STRUCTURE PARAMETERS OF SPADA

0.007 fault size		0.014 fault size	
Parameter	value	Parameter	value
Epochs	9000	*Epochs*	22000
Learning rates μ	3×10^{-3}	*Learning rates* μ	5×10^{-4}
Features size (nodes)	550, 500, 450	*Features size (nodes)*	600, 550, 550
Hidden layer	3	*Hidden layer*	3

C. Experimental Results

The experimental results of the PDA on the joint data set are shown in the Tables III and IV. For comparison purpose, the accuracy is also presented as a histogram in Fig. 8. It can be seen that the average accuracy of the SPADA method is 93.85% and 94.43% for the fault size 0.007 and 0.014, respectively, which are much higher than the other five compared methods. Specifically, we tested the classification accuracy of the six methods in the PDA scene and the normal DA scene, in order to make a better comparison. Note that the data in the CWRU dataset have obvious fault characteristics but with relatively lower of noise, so problem can be treated as a normal domain adaptation scenario, for which all the DA learning methods generally show high accuracy performance. For the joint dataset, SVM (without using DA method) shows poor performance in terms of classification accuracy in the entire PDA and normal DA, this implies that there is a domain shift between the source and target domain. On the other hand, when TCA and JDA methods, based on the predefined domain shift discrepancy, are used for data testing, good fault diagnosis results have been achieved in the normal DA, but in the PDA scenario, the classification accuracy declines seriously. DADA methods showed similar performance. This is mainly because the data of the gearbox data are transferred in the process of

Manuscript accepted (8 December 2020) for publication by *IEEE Transactions on Industrial Informatics*. DOI: 10.1109/TII.2020.3045002
IEEE-TII Early Access link: https://ieeexplore.ieee.org/document/9294067

TABLE III
TESTING ACCURACIES ON THE PDA TASKS ON 0.007 FAULT SIZE (%)

Method	SVM		JDA		DADA		SAN		MADA		SPADA	
PDA task	Partial	Non-partial	Partial	Non-partial	Partial	Non-partial	Partial	Non-partial	Partial	Non-partial	Partial	Non-partial
D1	49.14	77.21	49.09	83.28	52.50	94.12	82.81	96.37	83.47	92.32	**87.38**	**99.47**
D2	52.48	66.85	51.94	87.37	53.59	82.21	84.28	92.53	92.34	96.17	**92.59**	**99.29**
D3	57.58	80.66	76.94	92.79	59.72	84.09	94.97	97.46	86.24	**99.48**	**94.27**	99.36
D4	52.34	68.82	46.72	74.48	50.09	85.87	82.91	89.34	85.68	94.58	**86.73**	**98.78**
D5	52.48	60.20	47.63	77.19	60.72	89.42	87.50	93.22	87.37	96.14	**99.63**	**98.64**
D6	57.58	74.00	49.66	87.38	53.03	90.68	85.47	92.65	91.16	**100.00**	**98.44**	98.19
D7	54.35	67.43	51.49	88.35	50.38	80.22	81.82	97.79	84.72	86.24	**87.64**	**99.20**
D8	52.67	62.72	57.53	86.74	57.72	94.67	96.38	85.00	91.62	91.62	**98.13**	**98.87**
D9	62.08	70.83	53.38	90.26	54.62	80.75	86.61	96.38	96.47	97.45	**99.88**	**97.04**
Average	56.39	69.86	53.82	85.32	54.71	86.89	86.97	93.42	88.79	94.89	**93.85**	**98.76**

TABLE IV
TESTING ACCURACIES ON THE PDA TASKS ON 0.014 FAULT SIZE (%)

Method	SVM		JDA		DADA		SAN		MADA		SPADA	
PDA task	Partial	Non-partial	Partial	Non-partial	Partial	Non-partial	Partial	Non-partial	Partial	Non-partial	Partial	Non-partial
T1	51.27	74.13	52.68	78.43	54.25	98.61	81.41	97.17	86.05	94.25	**91.43**	**99.28**
T2	49.36	76.34	56.17	92.31	60.17	99.57	86.24	94.32	89.31	97.32	**94.24**	**99.36**
T3	56.19	73.21	68.23	86.84	58.46	96.35	90.32	96.60	90.27	96.60	**99.51**	**99.89**
T4	54.42	69.42	52.39	79.96	51.39	84.76	81.09	92.84	83.63	95.61	**89.20**	**100.00**
T5	50.09	66.06	59.51	76.47	58.92	91.42	88.73	95.51	88.42	94.57	**99.98**	**98.42**
T6	56.43	78.37	60.37	87.71	59.08	93.28	90.56	94.43	92.54	96.74	**92.65**	**99.96**
T7	58.96	68.58	54.72	86.82	53.76	84.46	86.18	98.05	87.28	99.98	**90.16**	**100.00**
T8	57.83	67.65	58.47	89.19	56.84	96.15	92.47	99.98	90.70	99.86	**97.78**	**100.00**
T9	68.37	72.98	56.83	92.25	64.73	94.22	87.85	**100.00**	93.96	**100.00**	**99.42**	98.74
Average	55.88	71.86	57.71	85.55	55.51	93.20	87.21	96.54	89.13	97.21	**94.93**	**99.52**

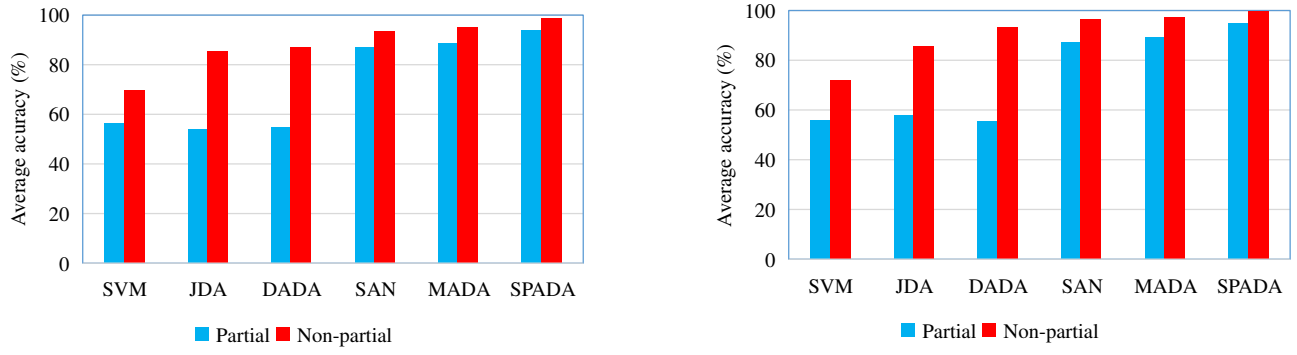


Fig. 8. Testing data classification accuracy histogram. (a) 0.007 fault size. (b) 0.014 fault size.

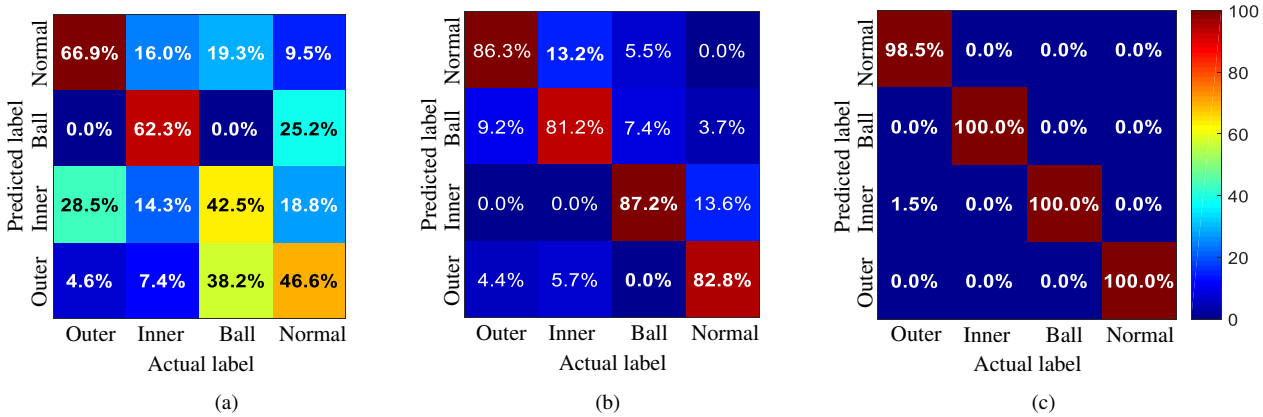


Fig. 9. The confusion matrices of the test data. (a) SVM. (b) JDA. (c) SPADA.

domain shift discrepancy correction, and there is a negative transfer between the source and target domain. Therefore, the existing DA methods cannot be simply directly used to solve PDA problems. This is especially true for the case here where a large amount of gearbox data are mixed in the source domain. It is easy to obtain fault vibration data in different rotating machines. However, these data contain different domains, using these data directly for domain adaptation can cause serious negative transfer, thus hinders the transfer of knowledge between domains.

Additionally, the proposed weighted domain discriminator technique was incorporated into SVM, JDA, and the proposed method. The confusion matrices relating to these three methods are shown in Fig. 9. Without using the weighted domain

Manuscript accepted (8 December 2020) for publication by *IEEE Transactions on Industrial Informatics*. DOI: 10.1109/TII.2020.3045002
IEEE-TII Early Access link: https://ieeexplore.ieee.org/document/9294067

classifier, DADA produces a much lower fault diagnosis accuracy than that of the proposed method in the PDA scenario. Moreover, SAN and MADA were also applied to the same data. A comparison of these methods are shown in Tables III and IV, where it can be seen that the proposed method is better than the two compared PDA methods in both normal DA scenarios and PDA scenarios. For instance, for the 0.007 fault size, the average accuracy of SPADA are 93.85% and 98.76% in the PDA and normal DA scene, but for SAN and MADA, the accuracies are 86.97%, 93.42% and 88.79%, 94.89%, respectively.

D. Empirical Analysis

The SAE neural network model, as a generator, mainly relies on its network structure to extract hidden features of the data, so the optimization of the network structure, together with a set of good hyper-parameters [32] is significantly important. Specifically, the number of hidden layer neurons, the learning rate, and the number of SAE network layers are the most important parameters that affect the training process and classification accuracy of the algorithm. In practice, the setting of the number of hidden layer neurons and the number of SAE network layers are inconclusive. It needs to be continuously adjusted according to the determined data set to achieve the best network layer-parameter. Taking the 0-1hp transfer situation as an example, the sample length is 600, so the range of the feature space is the interval [1, 600]. The experimental results are shown in Fig. 10, where it can be seen that a feature length of 400 samples is an appropriate choice for the case of 0.07 fault size. This implies that it is possible to achieve a higher accuracy using a relatively shorter length of samples. In other words, it is possible to obtain a satisfactory accuracy

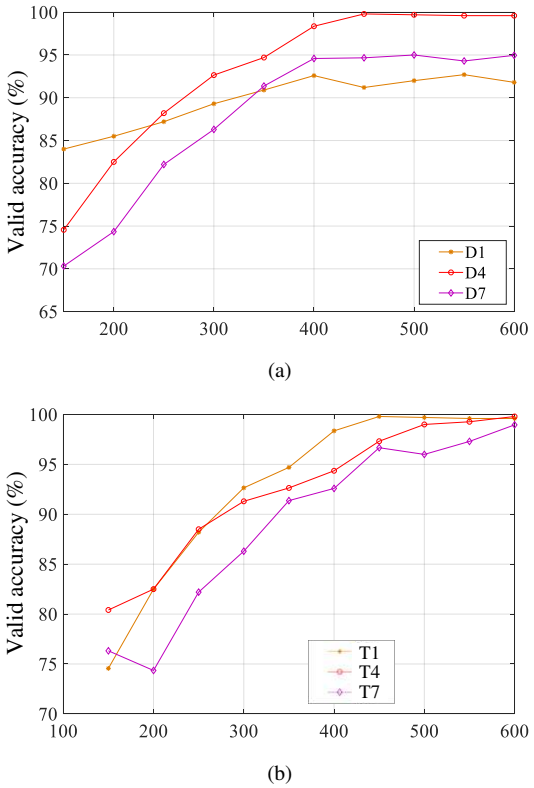


Fig. 10. The influence of feature size parameters on classification accuracy. (a) Feature space on 0.007 fault size. (b) Feature space on 0.014 fault size.

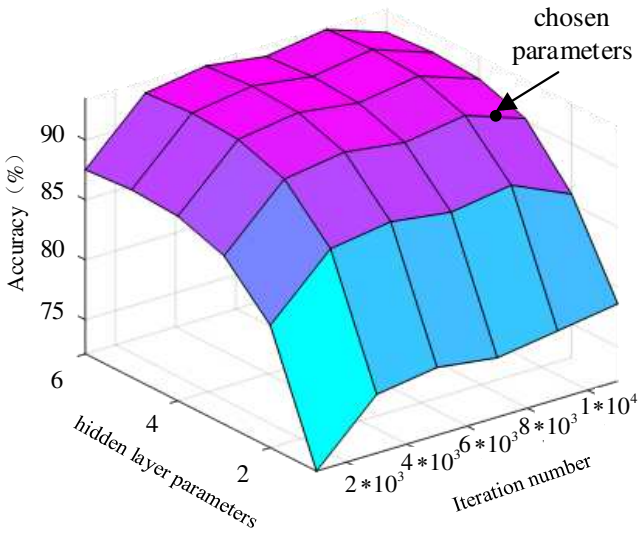


Fig. 11. Accuracy against the iteration and hidden layer parameters for the proposed method.

performance without significantly increasing computational load.

The number of SAE network layers is another main influencing factor. The maximum number of iterations is also the main factor influencing the performance of the model. The changes of these two factors on accuracy performance are shown in Fig. 11, which suggests that the number of hidden layers can be chosen as 3, and the number of iterations can be set to around 9×10^3.

E. Feature Visualization

To further verify the partial domain adaptation properties of the proposed method, we also computed the high-dimensional spatial alignment of the source and target class labels using the t-distributed stochastic neighbor embedding (t-SNE) [33]. t-SNE is a visualization tool that can be used to understand the local structure of high-dimensional data 2D or 3D space. An illustration of the t-SNE visualization for the results produced by the two methods SPADA (the proposed method) and JDA, for the case of 0.007 fault size, is given in Fig. 12. Note that in order to show the domain shift correction of the feature samples more clearly, the t-SNE features are separately displayed in the reduced dimensional space as shown in Fig. 12(a)-(d). Taking Fig. 12(a) as an example, for the proposed method, the distance between the normal features in source domain and target domain (from the purple × mark and blue × mark) is much closer than the that of JDA (from the red + mark and the green + mark). Similar observations can be obtained from Fig. 12(b)-(d), which display ball, inner, and outer features, respectively. These results illustrate the excellent domain adaptation ability of the SPADA method in the PDA scenarios.

F. Convergence Performance and Computational Time

Firstly, all the experiments are performed on the Nvidia 1050Ti and the Intel i5-7500 CPU. A higher performance CPU or GPU can reduce the algorithm running time.

As shown in Fig. 13, the test errors of the SPADA method converge fast in the three PDA tasks, and the SPADA converges to a satisfactory test error. Such a result implies that SPADA is not only efficient but also stable.

Manuscript accepted (8 December 2020) for publication by *IEEE Transactions on Industrial Informatics*. DOI: 10.1109/TII.2020.3045002
IEEE-TII Early Access link: https://ieeexplore.ieee.org/document/9294067

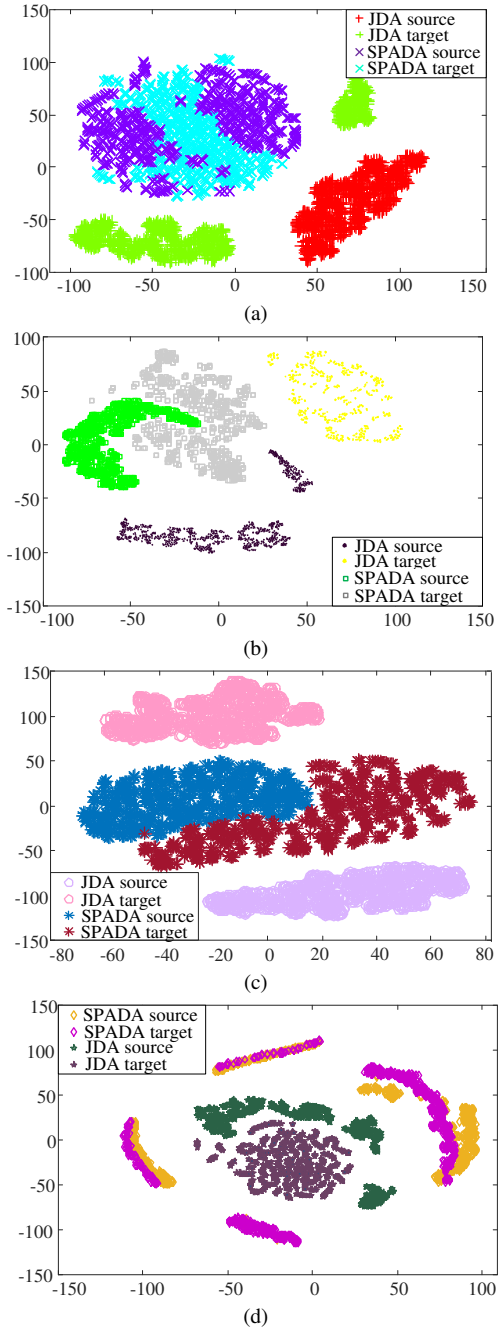


Fig. 12. A visual illustration of the four features revealed by the t-SNE dimensionality reduction approach. (a) Normal features (0.007 fault size). (b) Ball features (0.007 fault size). (c) Inner features (0.007 fault size). (d) Outer features (0.007 fault size).

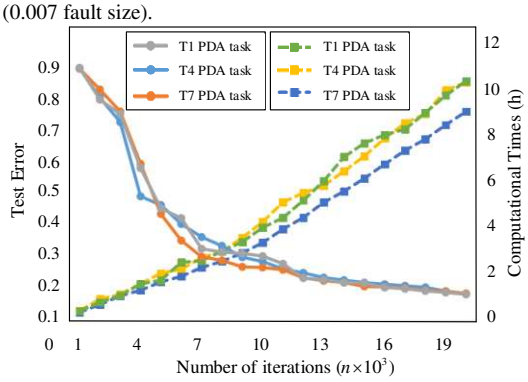


Fig. 13. The convergence performance and computational time against iterations of the SPADA method.

V. Conclusion

This paper proposed a new partial adversarial domain adaptation model (named SPADA) based on generative adversarial networks. The method was applied to fault diagnosis of rotating machines and tested using two benchmark datasets, where the fault types in the source sample space do not match those in the target domain. SPADA uses a SAE network to extract the representative features hidden in the original data. Then, it used a weighted domain classifier that can achieve sample weighting in the source domain data only, and used another domain classifier to implement domain adaptation. It finally used a label classifier to classify the target domain samples. Experiments show that the proposed SPADA algorithm can weight and select the common sample types from the mixed source domain samples. Comparison results show that SPADA outperforms the main existing methods. In this study, we select two fault sizes (0.007 and 0.014) and four different loads (0, 1, 2, and 3hp) to simulate the scenario for PDA, while the gearbox data are considered as anomalous classes to cause negative transfer only under one working condition (one shaft speed and one load). In order to better simulate the industrial process data, the influence of gearbox data under different shaft speeds and loads on the performance of PDA model will be further investigated in the future.

References

[1] Z. Gao, C. Cecati, and S. X. Ding, "A survey of fault diagnosis and fault-tolerant techniques—Part I: Fault diagnosis with model-based and signal-based approaches," *IEEE Trans. Ind. Electron.*, vol. 62, no. 6, pp. 3757-3767, Jun. 2015.

[2] S. Yin, J. Rodriguez-Andina, Y. C. Jiang, "Real-Time Monitoring and Control of Industrial Cyberphysical Systems: With Integrated Plant-Wide Monitoring and Control Framework," *IEEE Ind. Electron. Mag*., Vol.13, no.4, pp. 38 - 47, Dec. 2019.

[3] K. Worden, W. J. Staszewski, and J. J. Hensman, "Natural computing for mechanical systems research: A tutorial overview," *Mech. Syst. Signal Process.*, vol. 25, no. 1, pp. 4-111, Jan. 2011.

[4] Y. C. Jiang, S. Yin, "Recent Advances in Key-Performance-Indicator Oriented Prognosis and Diagnosis With a MATLAB Toolbox: DB-KIT," *IEEE Trans. Ind. Informat.*, Vol.15, no.5, pp.2849 - 2858, May. 2019.

[5] Z. Chen, G. He, J. Li, Y. Liao, K. Gryllias, and W. Li, "Domain adversarial transfer network for cross-domain fault diagnosis of rotary machinery," *IEEE Trans. Instrum. Meas.*, doi: 10.1109/TIM.2020.2995441, May. 2020.

[6] L. Shao, F. Zhu, and X. Li, "Transfer learning for visual categorization: A survey," *IEEE Trans. Neural Netw.*, vol. 26, no. 5, pp. 1019-1034, May. 2015.

[7] Z. Chai and C. Zhao, "A fine-grained adversarial network method for cross-domain industrial fault diagnosis," *IEEE Trans. Autom. Sci. Eng.*, vol. 17, no. 3, pp. 1432-1442, Jul. 2020.

[8] L. Guo, Y. Lei, S. Xing, T. Yan, and N. Li, "Deep convolutional transfer learning network: A new method for intelligent fault diagnosis of machines with unlabeled data," *IEEE Trans. Ind. Electron.*, vol. 66, no. 9, pp. 7316-7325, Sept. 2019.

[9] W. Lu, B. Liang, Y. Cheng, D. Meng, J. Yang, and T. Zhang, "Deep model based domain adaptation for fault diagnosis," *IEEE Trans. Ind. Electron.*, vol. 64, no. 3, pp. 2296-2305, Mar. 2017.

[10] W. Qian, S. Li, and J. Wang, "A new transfer learning method and its application on rotating machine fault diagnosis under variant working conditions," *IEEE Access*, vol. 6, pp. 69907-69917, 2018.

[11] Z. Tong, W. Li, B. Zhang, F. Jiang, and G. Zhou, "Bearing fault diagnosis under variable working conditions based on domain adaptation using Feature Transfer Learning," *IEEE Access*, vol. 6, pp. 76187-76197, 2018.

[12] Q. Guo, Y. Li, Y. Song, D. Wang, and W. Chen, "Intelligent fault diagnosis method based on full 1-D convolutional generative adversarial network," *IEEE Trans. Ind. Informat.*,vol. 16, no. 3, pp. 2044-2053, Mar. 2020.

Manuscript accepted (8 December 2020) for publication by *IEEE Transactions on Industrial Informatics*. DOI: 10.1109/TII.2020.3045002
IEEE-TII Early Access link: https://ieeexplore.ieee.org/document/9294067

[13] A. Chadha and Y. Andreopoulos, "Improved techniques for adversarial discriminative domain adaptation," *IEEE Trans. Image Process.*, vol. 29, pp. 2622-2637, 2020.
[14] B. Gholami, P. Sahu, O. Rudovic, K. Bousmalis, and V. Pavlovic, "Unsupervised multi-target domain adaptation: An information theoretic approach," *IEEE Trans. Image Process.*, vol. 29, pp. 3993-4002, 2020.
[15] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *J. Big Data.*, vol. 3, no. 1, pp. 9, 2016.
[16] F. Mahmood, R. Chen, and N. J. Durr, "Unsupervised reverse domain adaptation for synthetic medical images via adversarial training," *IEEE Trans. Med. Imag.*, vol. 37, no. 12, pp. 2572-2581, Dec. 2018.
[17] Y. Xie and T. Zhang, "A transfer learning strategy for rotation machinery fault diagnosis based on cycle-consistent generative adversarial networks," in *Proc. Chinese Automation Cong.*, pp. 1309-1313, 2018.
[18] J. Liu, F. Qu, X. Hong, and H. Zhang, "A small-sample wind turbine fault detection method with synthetic fault data using generative adversarial nets," *IEEE Trans. Ind. Informat.*, vol. 15, no. 7, pp. 3877-3888, Jul. 2019.
[19] P. Liang, C. Deng, J. Wu, G. Li, Z. Yang, and Y. Wang, "Intelligent fault diagnosis via semi-supervised generative adversarial nets and wavelet transform," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 7, pp. 4659-4671, Jul. 2020.
[20] X. Li and W. Zhang, "Deep learning-based partial domain adaptation method on intelligent machinery fault diagnostics," *IEEE Trans. Ind. Electron.*, doi: 10.1109/TIE.2020.2984968, Apr. 2020.
[21] W. Li, Z. Chen, and G. He, "A novel weighted adversarial transfer network for partial domain fault diagnosis of machinery," *IEEE Trans. Ind. Informat.*, doi: 10.1109/TII.2020.2994621, May. 2020..
[22] J. Jiao, M. Zhao, J. Lin, and C. Ding, "Classifier inconsistency-based domain adaptation network for partial transfer intelligent diagnosis," *IEEE Trans. Ind. Informat.*, vol. 16, no. 9, pp. 5965-5974, Sept. 2020.
[23] J. Li, X. Li, D. He, and Y. Qu, "A domain adaptation model for early gear pitting fault diagnosis based on deep transfer learning network," *J. Risk Relia.*, vol. 234, no.1, pp. 168-182, Feb. 2020.
[24] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, Y. Bengio,"Generative Adversarial Nets," *In Proc.* neural information processing systems, pages 2672–2680, 2014.
[25] K. Loparo. (2013). Case Western Reserve University Bearing Data Center. [Online]. Available: http://csegroups.case.edu/bearingdatacenter/pages/12k-drive-end-bearing-fault-data
[26] K. Goebel. (2009). The first Annual Conference of the Prognostics and Health Management Society[PHM09]. [Online]. Available: https://www.phmsociety.org/competition/PHM/09/
[27] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 415-425, Mar. 2002.
[28] M. S. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer feature learning with joint distribution adaptation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2200-2207.
[29] Z. Liu, B. Lu, H. Wei, L. Chen, X. Li, and M. Rätsch, "Deep adversarial domain adaptation model for bearing fault diagnosis," *IEEE Trans. Syst., Man, Cybern., Syst.*, doi: 10.1109/TSMC.2019.2932000.
[30] Z. Cao, M. Long, J. Wang, and M. I. Jordan, "Partial transfer learning with selective adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 2724-2732, 2018.
[31] Z. Pei, Z. Cao, M. Long, and J. Wang, "Multi-adversarial domain adaptation," in *Proc. Int. Conf. Assn. Art.*, pp. 2724-2732, 2018.
[32] L. Wen, L. Gao, and X. Li, "A new deep transfer learning based on sparse auto-encoder for fault diagnosis," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 49, no. 1, pp. 136-144, Jan. 2019.
[33] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9,no. 85, pp. 2579-2605, Nov. 2008.

Zhao-Hua Liu (M'16) received the M.Sc. degree in computer science and engineering, and the Ph.D. degree in automatic control and electrical engineering from the Hunan University, China, in 2010 and 2012, respectively. He worked as a visiting researcher in the Department of Automatic Control and Systems Engineering at the University of Sheffield, United Kingdom, from 2015 to 2016.

He is currently an Associate Professor with the School of Information and Electrical Engineering, Hunan University of Science and Technology, Xiangtan, China. His current research interests include artificial intelligence and machine learning algorithm design, parameter estimation and control of permanent-magnet synchronous machine drives, and condition monitoring and fault diagnosis for electric power equipment.

Dr. Liu has published a monograph in the field of *Biological immune system inspired hybrid intelligent algorithm and its applications*, and published more than 30 research papers in refereed journals and conferences, including IEEE TRANSACTIONS/JOURNAL/MAGAZINE. He is a regular reviewer for several international journals and conferences.

Bi-Liang Lu received the B.Eng. degree in electrical engineering and automation, the M.Sc. degree in automatic control and electrical engineering from the Hunan university of science and technology, Xiangtan, China, in 2017 and 2020, respectively.

His current research interests include deep learning algorithm design, and condition monitoring and fault diagnosis for electric power equipment.

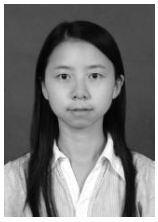
Hua-Liang Wei received the Ph.D. degree in automatic control from the University of Sheffield, Sheffield, U.K., in 2004.

He is currently a senior lecturer with the Department of Automatic Control and Systems Engineering, the University of Sheffield, Sheffield, UK. His research focuses on evolutionary algorithms, identification and modelling for complex nonlinear systems, applications and developments of signal processing, system identification and data modelling to control engineering.

Lei Chen received the M.S. degree in computer science and engineering, and the Ph.D. degree in automatic control and electrical engineering from the Hunan University, China, in 2012 and 2017, respectively.

He is currently a Lecturer with the School of Information and Electrical Engineering, Hunan University of Science and Technology, Xiangtan, China. His current research interests include deep learning, network representation learning, information security of industrial control system and big data analysis.

Manuscript accepted (8 December 2020) for publication by *IEEE Transactions on Industrial Informatics*. DOI: 10.1109/TII.2020.3045002
IEEE-TII Early Access link: https://ieeexplore.ieee.org/document/9294067



Xiao-Hua Li received the B.Eng. degree in computer science and technology from Hunan University of Science and Engineering, Yong zhou, China, in 2007 and the M.Sc. degree in computer science from Hunan University, Changsha, China, in 2010. Currently, She is currently a lecturer in the School of Information and Electrical Engineering, Hunan University of Science and Technology, Xiangtan, China. Her interests are in evolutionary computation.

Chang-Tong Wang received the B.Eng. degree in automation from the Hunan university of science and technology, Xiangtan, China, in 2019. He is currently pursuing the M.Sc. degree in automatic control and electrical engineering, at Hunan University of Science and Technology, Xiangtan, China.

His current research interests include deep learning algorithm design and wind power forecasting and dispatching.