
Article:
Lu, B. orcid.org/0000-0002-6023-295X, Zhang, Y. orcid.org/0000-0002-4170-6152, Liu, Z. et al. (2 more authors) (2023) A novel sample selection approach based universal unsupervised domain adaptation for fault diagnosis of rotating machinery. Reliability Engineering & System Safety, 240. 109618. ISSN 0951-8320

https://doi.org/10.1016/j.ress.2023.109618

Manuscript of the paper published in:
Reliability Engineering & System Safety, Volume 240, December 2023, 109618

A Novel Sample Selection Approach Based Universal Unsupervised Domain Adaptation for Fault Diagnosis of Rotating Machinery

Biliang Lu[a]; Yingjie Zhang[a]; Zhaohua Liu[b]; Hua-Liang Wei[c]; Qing Shuai Sun[a].

[a]*School of Computer Science and Electronic Engineering，Hunan University, Changsha 410000, PR China.*

[b]*School of Information and Electrical Engineering, Hunan University of Science and Technology, Xiangtan 411100, PR China.*

[C] *Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield S10, U.K.*

***Abstract*— Fault diagnosis in rotating machinery is a critical task within industrial settings. Unsupervised domain adaptation (UDA) methods have demonstrated significant potential in addressing the challenges of insufficiently labeled signal problems. However, the assumption that the label spaces of two domains are identical may not always be valid in real-world scenarios, thereby limiting the applicability of UDA. In this paper, we designed a more common UDA scenario, referred to as universal UDA (UUDA), to better handle domain and label space shift issues. Furthermore, we propose a novel sample selection method named Outlier Threshold Learning and Domain-Invariant Sampler (OTDI) to address the UUDA problem in fault diagnosis. Firstly, the outlier threshold learning aims to minimize the distance between known classes in the source domain while preserving the discrepancy between known and outlier classes. This enables the learning of a suitable and robust threshold for distinguishing known and outlier classes in the target domain. Subsequently, the domain-invariant sampler permits OTDI to execute domain-invariant feature sampling while accommodating label space shifts. Lastly, an adversarial classifier training method is incorporated to enhance transferability by acknowledging the variability of the label spaces across both domains. Extensive experiments were conducted on two datasets, and the proposed method demonstrated exceptional performance and superiority in managing domain and category inconsistencies.**

***Index Terms*—Machinery fault diagnosis; transfer learning; deep learning; unsupervised domain adaptation; universal unsupervised domain adaptation.**

1. Introduction

Fault diagnosis of rotating machinery is a crucial technology that ensures stable and safe operation in industrial production, notably within sectors such as manufacturing, energy, and transportation. It not only improves production efficiency and reduces maintenance expenses but also helps prevent safety accidents [1]. Traditional fault diagnosis technology, which typically uses signal processing methods to manually extract representative features, is often time-consuming, costly, or both [2]. However, with the increasing availability of machine condition monitoring data, deep learning-based methods are gaining more attention from both academia and industry due to their ability to automatically extract features and potential for improved accuracy.

Deep learning (DL) methods offer effective solutions that require little or no manual feature extraction [3]. Numerous DL methods have been introduced for intelligent fault detection, including convolutional neural networks (CNN) [4], stacked auto-encoders (SAE) [5], deep belief networks (DBN) [6], and deep residual shrinkage networks (DRSN) [7], among others. While these DL methods have demonstrated promising results in fault diagnosis, they present two primary limitations: 1) DL methods presume that the training and test sets follow the same distribution, an assumption difficult to fulfill in the current industry due to various uncertainties, such as different types of machinery, changing working conditions, and load variations across different scenarios. 2) DL methods require a substantial amount of labeled data, and model-free reinforcement learning often necessitates a high number of trials. Consequently, there remain unresolved challenges in DL methods that demand further exploration.

Consequently, domain adaptation (DA), and in particular, unsupervised domain adaptation (UDA), has been introduced to facilitate fault diagnosis under multiple or variable working conditions [8], [9]. UDA typically employs deep learning structures and seeks to minimize domain distances, aiming to extract domain-invariant features from both the source domain (SD) and the target domain (TD) and align any domain shifts. This approach aims to mitigate the effects of varying machinery types, loads, working environments, and other factors. For example, Yang et al. [10] proposed a deep multiple auto-encoder network with an attention mechanism to handle rotary machine fault diagnosis under different working

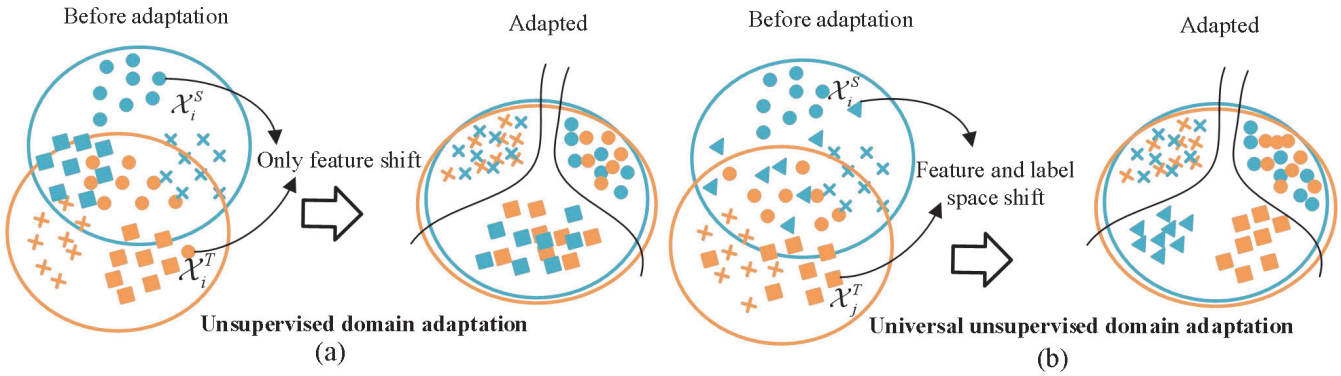


Fig.1. Domains (a) with only feature shift for UDA, and (b) with both feature shift and label shift for UUDA. In addition, all labels of the target domain are not available in the training phase.

conditions. Similarly, Ding et al. [11] developed a deep imbalanced domain adaptation network to address feature shift and label shift problems. However, most existing UDA research is primarily focused on closed-set scenarios, where the fault data paradigm remains constant, but differences exist in feature distribution [12], [13], as depicted in Fig. 1(a).

However, in practice, upholding the assumption of identical data paradigms between the SD and TD is not always feasible in industrial settings [14], [15], [16]. Real-world scenarios often present challenges such as the TD being a subset of the SD, which leads to a mismatch between the data paradigms. This situation is known as partial domain adaptation [17], [18], [19]. Furthermore, the emergence of new fault types during testing can result in the SD becoming a subset of the TD. This is referred to as open-set domain adaptation [20], [21], [22]. To address these challenges, researchers have proposed various Domain Adaptation (DA) techniques that require prior knowledge of the TD. For instance, Zhao et al. [23] introduced a dual adversarial network for cross-domain open-set fault diagnosis, while Ge et al. [24] presented an adaptive fault diagnosis method for rotating machinery with unknown faults under multiple working conditions. However, it's worth noting that these DA approaches are contingent upon prior knowledge of the TD, which may not always be available in practical scenarios. In some cases, we may only have monitoring data regarding the health of the device, without any prior information about the specific failures being diagnosed.

Therefore, to address the challenges posed by the universal unsupervised domain adaptation (UUDA) scenario, where labeled SD and unlabeled TD may present feature shifts along with label space shifts, without prior knowledge of the

differences. This paper extends the concept of UDA to universal UDA, as illustrated in Fig.1 (b), and we will elaborate on the problem setting of UUDA later in the section 3. The main challenges in UUDA can be summarized as follows: 1) Traditional DA methods for fault diagnosis primarily focus on matching feature shifts between domains, often overlooking potential differences in the label space. 2) Aligning the differences in both feature and label spaces becomes particularly challenging when the TD lacks labeled samples. 3) The model should possess the capability to classify target samples as "unknown" if they do not belong to any category present in the SD. To the best of our knowledge, fault diagnosis with domain shift under uncertain label space shift has not been extensively investigated, despite its relevance to engineering applications. To bridge this research gap, we propose a novel approach named the Outlier Threshold learning and Domain-Invariant sampler network (OTDI) to address the UUDA problem in fault diagnosis. Furthermore, we introduce a fault diagnosis method based on OTDI. The key contributions of our work can be summarized as follows:

1) We address the challenges of DA in fault diagnosis by introducing the concept of UUDA. This extends the traditional UDA methods to handle cases where the labeled SD and unlabeled TD demonstrate feature and label space shifts without prior knowledge of the differences.

2) We propose a novel sample selection approach, outlier threshold learning and domain-invariant sampler network (OTDI). This approach addresses the challenges of aligning differences in both feature and label spaces in UUDA scenarios, particularly when the TD lacks labeled samples.

3) To enhance d diagnostic accuracy under variable working conditions within the UUDA setting, we design an intelligent fault diagnosis algorithm based on OTDI. This algorithm incorporates an outlier threshold learning mechanism to automatically learn thresholds from the SD data, thus eliminating the need for manual validation or statistical methods. Moreover, a domain-invariant sampler, guided by the learned outlier threshold, is employed to extract domain-invariant features.

4) We provide detailed information on constructing UUDA datasets, including dataset characteristics and acquisition methods. Further, we describe the structure of the OTDI network, specify hyperparameters, and provide an outline of the

training scheme. Experimental results show the superior classification accuracy and generalization ability of our proposed algorithm within the UUDA setting.

The rest of this paper is as follows. The preliminary information is presented in Section II. In Section III, we provide a novel sample selection approach based on outlier threshold learning and domain-invariant sampler for fault diagnosis of rotating machines, as well as its optimal solution. Experimental results on different UUDA situations, the analysis of the convergence performance, and the ablation study are presented in Section IV. Finally, a summary is given in Section V.

2. Preliminary information

2.1. Unsupervised domain adaptation

Unsupervised domain adaptation is based on predefined distances between domains, such as maximum mean discrepancy (MMD) [25], Kullback-Leibler (K-L) divergence [26], and Wasserstein distance [27]. Specifically, Wen et al. [28] proposed a new deep transfer learning method using a three-layer sparse auto-encoder and MMD. These methods compress data into a high-dimensional feature space by feeding them into a deep structure and then minimizing these predefined distances such as MMD and KL divergence. Specifically, MMD defines the following metric to measure the global discrepancy between domains

$$d_{\mathcal{H}}(p,q) \triangleq \left\| \mathbb{E}_p[\phi(X^s)] - \mathbb{E}_q[\phi(X^t)] \right\|_{\mathcal{H}}^2, \tag{1}$$

where $\mathcal{H}$ is the reproducing kernel Hilbert space (RKHS) with a kernel $\phi()$.

So the domain shift between both domains is reduced in the high-dimensional space. But it brings up the problem that the data paradigm needs to be the same for both domains. This means that the type and number of faults need to be the same in both the training data and the test data. Such an assumption is difficult to meet in many real industry applications.

2.2 Class-space-shift domain adaptation

Imbalanced domain adaptation, particularly in cases characterized by class space shift, has been extensively explored to address the issue of class space inconsistency in large-scale real-world datasets. A significant challenge arises when training deep models using imbalanced datasets: distinguishing unknown categories within the class space. Discriminative models

trained on such data often struggle to recognize these categories, exhibiting a bias towards the labeled categories. For instance, partial Domain Adaptation (DA) primarily considers the distribution and class differences between the two domains, and some promising results have been achieved in this area [29]. Open-set DA, in contrast, addresses a more challenging scenario: identifying category differences when the target domain is unlabeled. Several open-set DA approaches have recently been developed. Zhang et al. [30] proposed an instance-level weighted mechanism to reflect the similarities of testing samples with known health states. Busto et al. [31] suggested an approach for open-set DA where the target domain contains instances of categories not present in the source domain. While these methods can achieve good performance in Partial DA (PDA) and Open-Set DA (OSDA) settings, they still rely on a priori knowledge of the Target Domain (TD).

3. Proposed method

In this section, we first give the problem description for UUDA in 3.1. In addition, the outlier threshold learning idea will be described in 3.2. The domain shift will be aligned through the domain-invariant feature sampler, as shown in 3.3. Then, 3.4. provides the overall training and loss function of OTDI, which is shown in Algorithm 1. Finally, the detailed workflow of the diagnostic framework can be found in 3.5.

3.1. Problem description

Given a source domain D_S with the task $\mathcal{T}_S$, and a target domain D_T with the task $\mathcal{T}_T$, UUDA is concerned with learning to a prediction function $f_T(.)$ on the D_T by using information on D_S and $\mathcal{T}_S$, where $D_S \neq D_T$ or $\mathcal{T}_S \neq \mathcal{T}_T$, and labels in $\mathcal{Y}_S$ are known, but $\mathcal{Y}_T$ is not observable. Meanwhile, the commonness between the SD and TD label spaces is defined as $C = \frac{\mathcal{Y}_S \cap \mathcal{Y}_T}{\mathcal{Y}_S \cup \mathcal{Y}_T}$. In UUDA, $C < 1$, and a smaller C implies a larger category shift. So, the task for UUDA is to learn a classification model f to minimize the target risk in the common label set, i.e. $\min \mathbb{E}_{(x,y)\sim D_C}[f(x) \neq y]$.

In the above definition, $D_S \neq D_T$ means $\mathcal{X}_S \neq \mathcal{X}_T$ or $P_S(X) \neq P_T(X)$, and $\mathcal{T}_S \neq \mathcal{T}_T$ implies $\mathcal{Y}_S \neq \mathcal{Y}_T$ or $P(y_S|x_S) \neq P(y_T|x_T)$. For the commonness, C=1 means that $\mathcal{Y}_S \cap \mathcal{Y}_T = \mathcal{Y}_S \cup \mathcal{Y}_T$, and the DA scenario is the traditional closed-set domain adaptation.

3.2. Outliers Threshold Learning

Our novel idea involves training multiple classifiers to learn the boundaries between known and outlier values for each category in a fine-grained way. A clear structure is shown in Fig.2. For a given input sample, the goal is to enable the classifier to learn the distance between the known and outlier classes. This distance should serve as an effective threshold for rejecting "unknown" samples. This process can be achieved as follows: For each known sample, its own label can be used to train the classifier. Simultaneously, a class-invariant space should also be defined and determined, which enables the formation of an effective decision boundary between known samples and outlier samples. During the outlier threshold learning phase, a linear classifier (which we treat as a sub-classifier) is introduced for each class. The objective of each sub-classifier is to compute the probabilities of known and outlier classes for the sample features.

Specifically, given the input features $z_i = G_\theta(x_i)$, where G_θ and x_i represents the feature extractor and $i-$th input sample, respectively, and z_i denotes the representative features extracted by G_θ . The likelihood of each class can be obtained by linearly weighting the samples. Thus, the score of z_i belonging to a certain class k can be expressed as:

$$c_i^k = \omega_k z_i \in \mathbb{R}^2 \tag{2}$$

where c_i^k contains the probability of belonging to the known and outlier classes in k category. Let $p(y_k \,|x_i) = f(c_i^k)$ denote the probability that the input sample x_i belongs to the known class k , where $f(.)$ is the softmax activation function, then the value of each input sample is normalized to [0, 1].

Now, we can define the loss function of outlier threshold learning on the input samples (x_i, y_i) as:

$$\mathcal{L}_{otl}(x_i, y_k) = \frac{1}{n}\sum_{i=1}^{n} -\log(p(y_k \,|x_i)) - \min_{j \neq y_k} \log(1 - p(y_j \,|x_i)) \tag{3}$$

Where $\min_{j \neq y_k}$ means the minimization is taking in the class-invariant space class. For a set of given input samples, equation (3) defines the loss on a known and the class-invariant space class, this is referred to as outlier threshold learning (OTL).

3.3. Domain-Invariant Feature Sampler

The key of the domain-invariant feature sampler is the separation of outlier samples from the two domains. Our approach aims to boost the confidence of common category alignment for adversarial DA. To achieve this, the learned outlier rejection threshold is applied to each sample during adversarial DA for each sample $x_t \in \mathcal{D}_t$ to train the classifiers. After the training,

the OTL classifiers establish a decision boundary for the “outliers” classes. Following this, We calculate the class probability of all classifiers for sample x_i and get the probability that x_i belongs to the outlier. More specifically, given the multiple classifiers, and the ω_t can be easily obtained as,

$$\omega_t(g(x_i^{s,t})) = p(y=1 \mid g(x_i^{s,t})) = \frac{e^{x_i^{s,t}}}{\sum_{j=1}^{K} e^{x_j^{s,t}}} \tag{4}$$

Where K is the number of classes in the multi-class classifier, $y=1$ suppose the probability that a classified feature is labeled as known class is 1, then $\omega_t(g(x_i^{s,t})) \approx 1$. The outlier samples do not participate in the common subspace adaptation and will be filtered out as follows:

$$label_{otl} = \begin{cases} Known & \text{if } \omega_t > 0.7 \\ Unknown & \text{if } \omega_t < 0.3 \end{cases} \tag{5}$$

Thus, obvious outlier samples are simply filtered out by a learned threshold and can be subjected to adversarial DA. The general objective $\mathcal{L}_{ada}$ to be optimized in adversarial DA can be defined as:

$$\begin{aligned} \min_{g(x_i^s), g(x_i^t)} \max_{D} \mathcal{L}(D, g(x_i^s), g(x_i^t)) = {} & \mathbb{E}_{x \sim p_s(x)}[\log D(g(x_i^s))] \\ & + \mathbb{E}_{x \sim p_t(x)}[\log(1 - D(g(x_i^t)))] \end{aligned} \tag{6}$$

Where D denotes the domain classifier. $\mathbb{E}$ is the mathematical expectation. For $0.3 \le \omega_t^*(g(x_i^{s,t})) \le 0.7$, it is uncertain to determine whether the both domain samples are known or outliers. Hence, we weight the adversarial domain adaptation with the learned threshold. The features weighting function should be inversely proportional to $\omega_t^*(g(x_i^{s,t}))$ since the unlabeled target domain samples need to be given a more significant weight; the entire weighting process can be written as follows:

$$\widetilde{\mu}(g(x_i^t)) = 1 - \omega_t^*(g(x_i^{s,t})) \tag{7}$$

Given $g(x_i^s)$,for any $g(x_i^t)$, the optimum $\omega_t^*(g(x_i^{s,t}))$ is obtained at:

$$\omega_t^*(g(x_i^{s,t})) = \frac{p_s(g(x_i^{s,t}))}{p_s(g(x_i^{s,t})) + p_t(g(x_i^{s,t}))} \tag{8}$$

Where $g(x_i)$ is the representative features after feature extraction networks. The proof of (8) is given as follows.

Proof: For any $g(x_i^s)$ and $g(x_i^t)$, the domain classifier D is to maximize (8)

$$\begin{aligned}&\max_{D}\mathcal{L}(D,g(x_i^s),g(x_i^t))\\&=\int_x p_s(x)\log D(g(x_i^s))+p_t(x)\log(1-D(g(x_i^t)))dx\\&=\int_{g(x_i^{s,t})} p_s(g(x_i^{s,t}))\log D(g(x_i^{s,t}))+p_t(g(x_i^{s,t}))\log(1-D(g(x_i^{s,t})))dg(x_i^{s,t}).\end{aligned}\tag{9}$$

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,(8) can be rewritten as:

$$\widetilde{\mu}(g(x_i^t))=1-\omega_t^*(g(x_i^{s,t}))=\frac{1}{\frac{p_s(g(x_i^{s,t}))}{p_t(g(x_i^{s,t}))}+1}\tag{10}$$

It can be seen that $\widetilde{\mu}(g(x_i^t))$ will become small if the ratio $\frac{p_s(g(x_i^{s,t}))}{p_t(g(x_i^{s,t}))}$ will be large. Therefore, the weight of the SD and TD samples from the outlier class will be less than the weight of the shared class. The objective function for domain-invariant feature sampler can be rewritten as:

$$\mathcal{L}_{DIFS}=\widetilde{\mu}(g(x_i^t))*\mathcal{L}_{ada}\tag{11}$$

3.4. Objective function and optimization strategy

With the domain-invariant features sampler obtained in the previous section, UUDA can be achieved by adversarial learning based on the common category, and it is able to reject outlier samples and achieve distribution discrepancy matching between SD and TD. In this section, we combine outlier threshold learning and the domain-invariant feature sampler to separate and classify outliers and known samples. For the known classifier, we can easily use a linear classifier to solve the classification problem by defining the cross entropy loss function $\mathcal{L}_{cls}(X_i^s,y_i^s)$.

$$\mathcal{L}_{cls}(X_i^s,y_i^s)=-\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{M}y_i^s\log(f(X_i^s))\tag{12}$$

Thus, the entire loss function can be written as follows:

$$\mathcal{L}_{all}=\mathop{\mathbb{E}}_{(x_i^s,y_i^s)\sim\mathcal{D}_s}\mathcal{L}_{OTL}(X_i^s,y_i^s)+\lambda\mathop{\mathbb{E}}_{(x_i^{s\,t},y_i^{s\,t})\sim\mathcal{D}_{s\,t}}\mathcal{L}_{DIFS}(X_i^{s\,t},y_i^{s\,t})\tag{13}$$

$$\mathcal{L}_{OTL}=\mathcal{L}_{cls}(X_i^s,y_i^s)+\mathcal{L}_{otl}(X_i^s,y_i^s)\tag{14}$$

Where $\mathcal{L}_{otl}(X_i^s,y_i^s)$ is defined by (3), $\mathcal{L}_{DIFS}(X_i^{s,t},y_i^{s,t})$ is defined by (11), and $\mathcal{L}_{OTL}$ is the entire loss of outliers threshold learning. The symbol $\mathbb{E}$ means to take the expectation of the associated variable. Then, the optimized parameters for network θ_f, label classifier θ_y, outlier classifier θ_{otl}, and domain classifier parameters θ_d can be computed as follows:

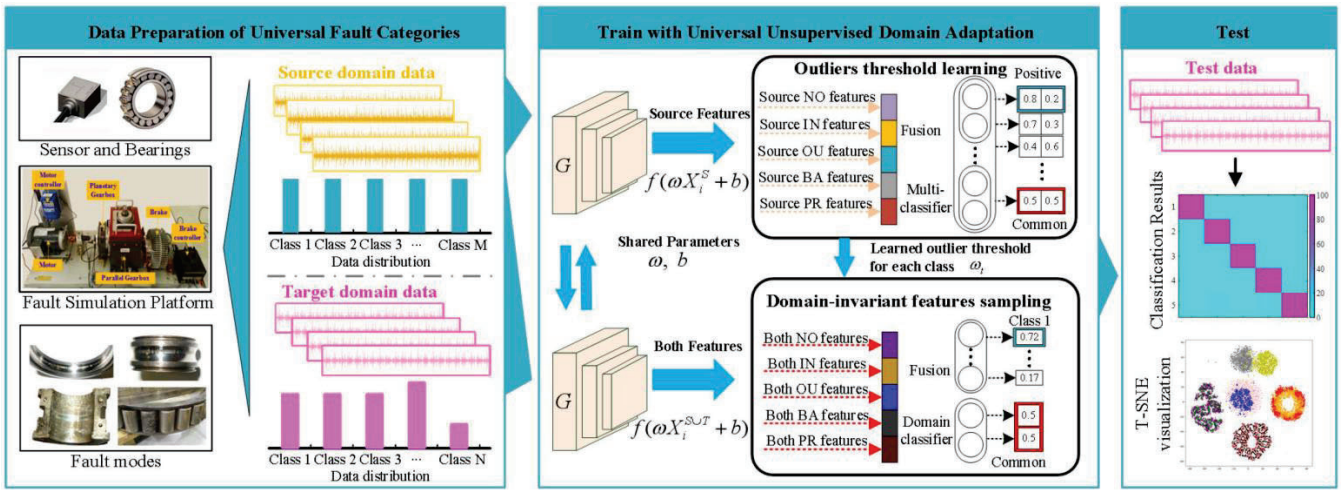


Fig.2. The workflow details of the proposed UUDA-based fault diagnosis method.

Algorithm 1: Algorithm Framework of OTDI

Input: The training set: *X_signal, Y_signal and X_label*

Output: The testing features label: *Y_label*

Begin

Batch_size ← 64;

Pre-training to initialize the OTDI parameters θ;

for epoch $i = 1$ **to** N **do**

  for k **in** *Batch_size* **do**

    $\{f_i\}_{i=1}^{n^s+n^t} \leftarrow 1DCNN\left(x_s \cup x_t\right)$;

  end

  Obtained the training and testing set features;

  while $i < n$ **do**

    The sample features $\{F_i\}_{i=1}^{n^s+n^t}$ are input to outlier threshold and domain-invariant sampler.

    for k **in** $\{\mathcal{L}_{cls}^{s}, \mathcal{L}_{otl}^{s}, \mathcal{L}_{DIFS}^{s,t}, \mathbf{1DCNN}\}$ **do**

      Update network parameters θ_f, label classifier θ_y, outlier classifier θ_{otl}, and domain classifier parameters θ_d according to (17)-(20).

    end

  end

end

end

Return unlabeled testing features corresponding *Y_label*.

TABLE I

DETAILED CONFIGURATION OF 1DCNN ARCHITECTURE DESCRIPTION

Layer	Type	Receptive field size - channels - stride - padding	Output
Input	*Data*	-	1024*1
Block1-conv1	*Convolution*	64*1 - 16 - 1 - 32	1024*16
Block1-pool1	*MaxPooling*	2*1 - 1 - 2	512*16
Block1-BN1	*Batch normalize*	-	512*16
Block1-ReLU1	*ReLU*	-	512*16
Block2-conv2	*Convolution*	5*1 - 32 - 1 - 2	512*32
Block2-pool2	*MaxPooling*	2*1 - 1 - 2	256*32
Block2-BN2	*Batch normalize*	-	256*32
Block2-ReLU2	*ReLU*	-	256*32
Block3-conv3	*Convolution*	5*1 - 64 - 1 - 2	256*64
Block3-pool3	*MaxPooling*	2*1 - 1 - 2	128*64
Block3-BN3	*Batch normalize*	-	128*64
Block3-ReLU3	*ReLU*	-	128*64
Block4-conv4	*Convolution*	5*1 - 64 - 1 - 2	128*64
Block4-pool4	*MaxPooling*	2*1 - 1 - 2	64*64
Block4-BN4	*Batch normalize*	-	64*64
Block4-ReLU4	*ReLU*	-	64*64
Fc1	*Fully-connected*	1*1*1024	1024
Fc2	*Fully-connected*	1*1*1024	1024
Output	*Fully-connected*	1*1*C	*C*

$$(\tilde{\theta}_f, \tilde{\theta}_y, \tilde{\theta}_{otl}) = \arg\min_{\theta_y, \theta_d} \mathcal{L}^s_{cls} + \mathcal{L}^s_{otl} + \mathcal{L}^{s,t}_{DIFS}, \tag{15}$$

$$(\tilde{\theta}_d) = \arg\min_{\theta_f} \mathcal{L}^s_{cls} + \mathcal{L}^s_{otl} - \mathcal{L}^{s,t}_{DIFS}. \tag{16}$$

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

$$\theta_f \leftarrow \theta_f - \mu\left(\partial\mathcal{L}_{cls}^{s} \setminus \partial\theta_y + \partial\mathcal{L}_{otl}^{s} \setminus \partial\theta_f - \alpha(\partial\mathcal{L}_{DIFS}^{s,t} \setminus \partial\theta_d)\right), \tag{17}$$

$$\theta_y \leftarrow \theta_y - \mu\left(\partial\mathcal{L}_{cls}^{s} \setminus \partial\theta_y + \partial\mathcal{L}_{otl}^{s} \setminus \partial\theta_f\right), \tag{18}$$

$$\theta_{otl} \leftarrow \theta_{otl} - \mu(\partial\mathcal{L}_{otl}^{s} \setminus \partial\theta_{otl}), \tag{19}$$

$$\theta_d \leftarrow \theta_d - \mu(\partial\mathcal{L}_{DIFS}^{s,t} \setminus \partial\theta_d). \tag{20}$$

Where μ is the learning rate, α is the weighting factor, $\partial\mathcal{L}_{cls}^{s} \setminus \partial\theta_y$, $\partial\mathcal{L}_{otl}^{s} \setminus \partial\theta_f$, $\partial\mathcal{L}_{otl}^{s} \setminus \partial\theta_{otl}$, and $\partial\mathcal{L}_{DIFS}^{s,t} \setminus \partial\theta_d$ are the partial derivatives of the loss function to the $\tilde{\theta}_f$, $\tilde{\theta}_y$, θ_{otl}, and $\tilde{\theta}_d$ parameter. Note that the gradient reversion in (17) (indicated by $-\alpha$) is performed between the generator G and the domain discriminator connection layer, and this is named as the gradient reversal layer (GRL).

The pseudo code of the above optimization process is shown in Algorithm 1.

3.5. OTDI for intelligent fault diagnosis

Applying OTDI to intelligent fault diagnosis, we use 1D-CNN as its backbone with the specific parameters shown in Table 1, and training samples from multiple time-domain vibration signals measured by accelerometers. Specifically. The diagnosis process is implemented in the following steps.

(1) Dataset setup: Collects vibration signals with different failure modes from multiple domains to create labeled source domain $\mathcal{D}_S$, unlabeled target domain $\mathcal{D}_T$ and unlabeled test sets $\mathcal{D}_{Test}$.

(2) Network parameters initialization: Build a deep network h_θ parameterized with random θ, which contains a 1D-CNN backbone $g(.)$ and a linear classifier $f(.)$.

(3) OTDI training: Train the initialized h_θ with datasets $\mathcal{D}_S$ and $\mathcal{D}_T$ using the OTDI algorithm.

(4) Test diagnosis performance: Input the test dataset to the OTDI with the optimized h_θ^*. Output the diagnosis result.

This workflow details of OTDI framework for fault diagnosis is also show in Fig.2.

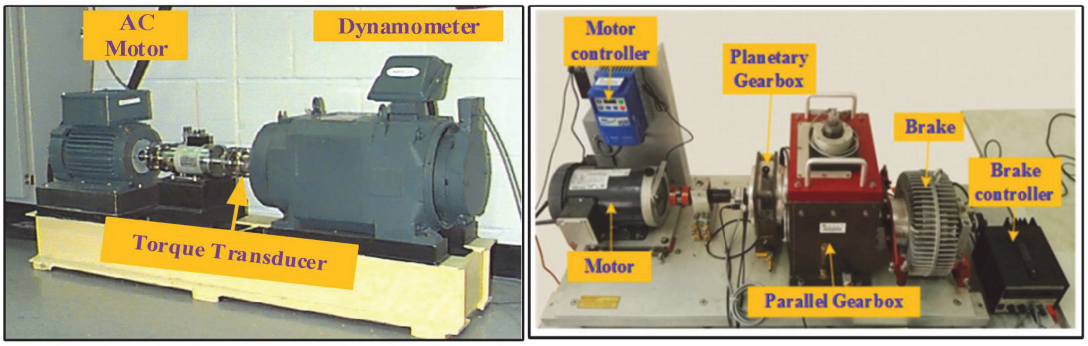
4. Experimental Test

Rolling bearings and planetary gearboxes are important mechanical devices in industrial systems; such equipment has a high failure rate. Therefore, in this section, we use two experimental datasets, collected from a rolling bearing and a gearbox,

respectively, to demonstrate the performance of the proposed methods.

4.1. *Experimental Setup*

4.1.1 Rolling Bearing Test Dataset



(a) Rolling bearing rig **(b) Planetary gearbox experiment platform**

Fig.3. Two dataset acquisition platforms.

The dataset is provided by the Case Western Reserve University (CWRU) Bearing Data Center [32]. The experimental equipment is shown in Fig.3 (a). The experimental data used in this paper was collected from the drive end bearing at a sampling frequency of 12KHz; it contains normal (N) signals and three types of faults, namely inner race fault (IF), outer race fault (OF) and roller element fault (RF). For the dataset, a total of twelve UUDA scenarios (named Task 1 - Task 12) are designed. Specifically, for task 1 (A-C(N)→B-D(IF)), A-C(N) is source domain, B-D(IF) is target domain, and A denotes the common set, C(N) denotes the private set (outliers), a similar situation is found in the target domain. More details are shown in Table III.

TABLE III

TWELVE DIAGNOSIS TASKS OF ROLLING BEARING

Domain	Motor loads	Health conditions	Diagnosis tasks (Indices task 1-12, from left to right, top to bottom.)
A	0 HP	N/IF/OF/RF	A-C(N)→B-D(IF), A-B(N)→C-D(IF), A-B(N)→D-C(IF)
B	1 HP	N/IF/OF/RF	B-C(N)→A-D(OF), B-A(N)→C-D(OF), B-A(N)→D-C(OF)
C	2 HP	N/IF/OF/RF	C-B(N)→A-D(RF), C-A(N)→B-D(RF), C-A(N)→D-B(RF)
D	3 HP	N/IF/OF/RF	D-B(N)→A-C(IF), D-A(N)→B-C(IF), D-A(N)→C-B(IF)

4.1.2 Planetary Gearbox Test Dataset

These data were collected from Drivetrain Dynamic Simulator (DDS) [33]; the associated experimental equipment is shown in Fig.3 (b). In general, the dataset includes gearbox and bearing data. For the gearbox, there are two domains with the change of the motor speed, and the load configuration scenarios can be categorized as either A→W or W→A, as shown in Table IV. In addition, each domain contains normal (N) data and four types of fault signals (failures), namely, chipped tooth fault (CF), miss tooth fault (MF), root tooth fault (RF), surface tooth fault (SF). In this paper, there are six diagnostic tasks for planetary gearboxes from A to W, Similarly, the planetary gearbox from W to A also has six diagnostic tasks. More details about the common and private set are shown in Table V.

TABLE IV

DIFFERENT FAULTS AND LOADS OF PLANETARY GEARBOX

Domain	Motor loads	Health conditions
A	20 HZ-0V	N/CF/MF/RF/SF
W	30 HZ-2V	N/CF/MF/RF/SF

TABLE V

SIX DIAGNOSIS TASKS FOR PLANETARY GEARBOX FROM A to W

Scenario index	Training set		Testing set	
	Common set	Private set	Common set	Private set
1	A(N,CF,MF)	A(RF)	W(N,CF,MF)	W(SF)
2	A(N,CF,RF)	A(MF)	W(N,CF, RF)	W(SF)
3	A(N,CF,SF)	A(MF)	W(N,CF,SF)	W(RF)
4	A(N,MF,RF)	A(CF)	W(N,MF,RF))	W(SF)
5	A(N,MF,SF)	A(CF)	W(N,MF,SF)	W(RF)
6	A(N,RF,SF)	A(CF)	W(N,RF,SF)	W(MF)

4.2. Comparison Methods and Application Details

In order to verify the performance of the OTDI algorithm, we used the above two datasets to design a variety of UUDA scenarios, which are described in the previous section. For comparison purposes, the following five state-of-the-art methods are also applied to all these UUDA scenarios: Support Vector Machine (SVM) [34], Deep Convolutional Neural Network (DCNN) [35], Domain-Adversarial Training of Neural Networks (DANN) [13], CORAL [36], Open Set Domain Adaptation by Back-Propagation (OSBP) [37] and Universal Adaptation Network (UAN) [38]. More specifically, the experimental settings and details of these methods and the OTDI are summarized as follows:

1) **SVM**: this is a traditional machine learning approach. Gaussian kernel function was used and the experiments were carried out using the Scikit-Learn. For all the adjustable parameters, we used the default values suggested by the software.

2) **DCNN**: this uses the OTDI method but without the outlier threshold and domain-invariant learning. It is a backbone for the OTDI. Specific detailed parameters are shown in Table I.

3) **CORAL**: this is a popular baseline method based on correlation alignment, and it minimizes domain-shift by aligning the second-order statistics of source and target distributions. It does not have any hyper-parameters.

4) **DANN**: this is committed to achieving distribution alignment through an adversarial training method. Specifically, the domain discriminator cannot distinguish whether the input samples belong to SD or TD, so that SD and TD become aligned in distribution. It has the same feature extractor as DCNN.

5) **OSBP**: this is an open-set DA method based on adversarial domain adaptation. It mainly solves the domain adaptation problem when the source domain is a subset of the target domain. The main optimization parameters are given in [37].

6) **UAN**: It is implemented in PyTorch and the DCNN is used as the backbone network. The pseudo-label weighting ω_s and ω_t are normalized to be within interval [0, 1] in a mini-batch. More details can be found in [38].

7) **OTDI**: the details about the main structure parameters are shown in Table I. In addition, the number of iterations and the mini-batch are set as 100 and 64, respectively, in the initialization stage. After initialization, the learning rate and the maximum iteration number are adjusted to 0.001 and 500, respectively, and the hyper-parameter λ is set to 0.45.

In summary, all experimental tests are trained on a Windows Desktop with 32GB RAM, and the Python version is 3.6 with the Pytorch 1.7.1, the central processing unit (CPU) and graphic processing unit (GPU) are i9-10900K and GTX 3090, 24GB VRAM, respectively. In addition, the computer unified device architecture (CUDA) is selected as 11.0, since the lower version of CUDA does not support the calculation of GTX 3090.

4.3. Rolling Bearing Test Results

4.3.1 Classification Accuracy Results

In this section, we applied the seven methods mentioned in the previous section to 12 UUDA scenarios to verify OTDI performance; these methods include machine learning, deep learning, traditional DA , and class-space-shift DA methods. The overall classification results, shown in Table II and Fig.4, indicate that the proposed OTDI model consistently provides highly accurate classification performance across various UUDA environments, surpassing all other methods. In particular, the classification accuracy of OTDI significantly exceeded that of the Support Vector Machine (SVM), averaging approximately 30% higher. Deep Convolutional Neural Network (DCNN) achieved an average accuracy of 82.07%, which is markedly

TABLE II

TESTING ACCURACIES OF ROLLING BEARING ON THE DIFFERENT Universal UDA TASKS (%)

Method	Task1	Task 2	Task 3	Task 4	Task 5	Task 6	Task 7	Task 8	Task 9	Task 10	Task 11	Task 12	**Average**
SVM	64.12	57.34	51.64	77.76	72.86	66.42	67.22	74.92	67.84	72.38	72.22	72.22	68.08
CORAL	58.76	51.82	55.78	59.30	50.08	52.42	56.10	46.66	41.28	42.58	40.24	41.80	49.74
DCNN	71.10	83.92	88.47	90.84	89.65	75.98	86.59	77.63	74.97	87.44	81.15	77.05	82.07
DANN	81.31	77.43	83.35	86.76	79.32	81.55	71.23	68.34	85.31	88.51	73.19	60.78	78.09
OSBP	88.56	93.12	84.78	83.14	80.78	79.66	85.84	85.14	83.42	87.24	85.58	83.96	85.10
UAN	92.80	96.06	91.62	89.64	79.12	77.38	85.10	91.16	87.54	88.00	86.48	84.08	87.42
OTDI	**99.08**	**99.90**	**99.10**	**98.62**	**99.80**	**99.52**	**93.10**	**92.54**	**96.84**	**92.76**	**91.06**	**95.06**	**96.45**

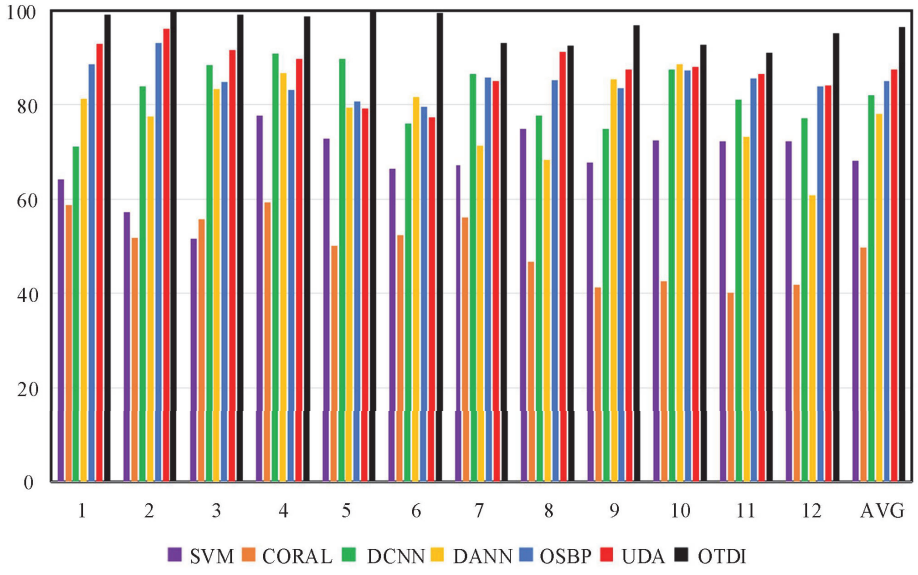


Fig.4. The accuracy of 12 tasks (indices 1-12) and average performed (AVG) on rolling bearings.

lower than OTDI. As for the traditional DA method, CORAL, it posted an accuracy rate of 49.74%, while the Domain-Adversarial Neural Network (DANN) achieved 78.09%, lower than DCNN without DA. Among the state-of-the-art DA methods, our proposed model outperformed others, achieving a performance gain of 7.4% and 4.1% over OSBP and UAN, respectively. Comparing with the classical SVM and CORAL, other diagnosis models have higher accuracy in the unlabeled TD. From Fig.5, we know that supervised learning methods can only generalize to known categories. Thus, it is not possible to diagnose outlier categories. Even traditional DA methods can even lead to negative transfer. The average diagnostic accuracy of DCNN is 82.07%, which is higher than the traditional DA, such as CORAL and DANN. This may be explained that the traditional DA approach produces a negative transfer effect in UUDA, since it is impossible to guarantee the separation of private categories while aligning public features.

For OSBP and UDA, the former uses a pseudo-label weighting approach to achieve the separation of outliers, so as to align only the common samples, and the latter uses an outlier threshold which is determined based on empirical values. However, the accuracy performance of OSBP and UDA in each transfer task is obviously lower than the OTDI model. This indicates that the outlier threshold and the domain-invariant feature sampler methods of OTDI are superior to empirical selection and pseudo-label weighting approaches.

Finally, we can draw the following conclusion: 1) The fault diagnosis results demonstrate that the proposed OTDI model

exhibits highly accurate classification performance across different UUDA environments, outperforming all other methods. 2) The analysis in Fig. 5 indicates that supervised learning methods are limited to known categories and unable to diagnose outlier categories effectively. Additionally, traditional DA methods may lead to negative transfer effects. In contrast, the OTDI model demonstrates higher accuracy, attributed to its outlier threshold and domain-invariant feature sampler methods, which outperform empirical selection and pseudo-label weighting approaches used in other methods like OSBP and UDA.

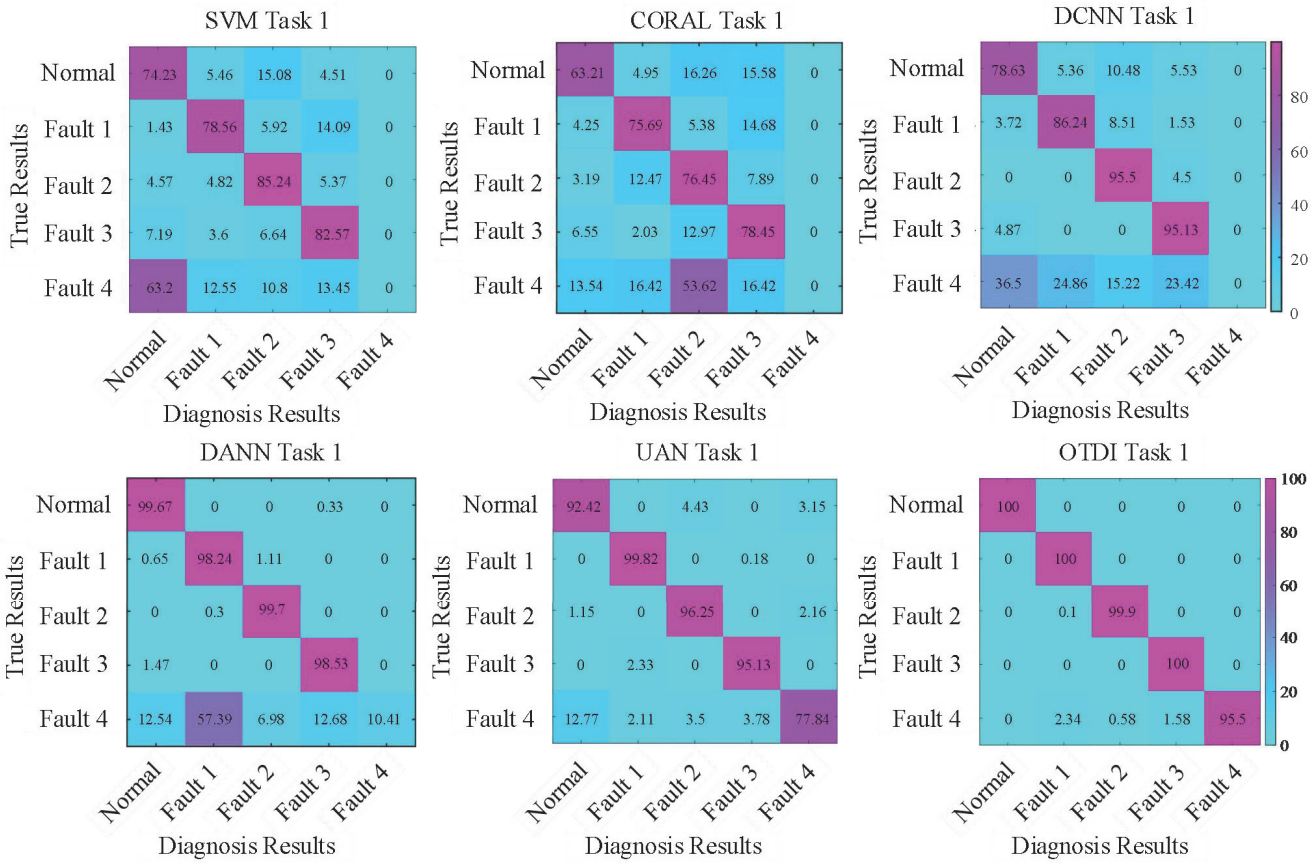


Fig.5. Confusion matrices of the six approach for bearing in Task 1.

4.3.2 Evaluation via Feature Visualization

Feature visualization is an essential tool for deciphering the performance of diagnosis models. In this section, we use the t-distributed stochastic neighbor embedding (t-SNE) method [34] to map high-dimensional features into a 2-D feature space. We randomly selected task 1 as the subject of our feature visualization, with detailed results shown in Fig.6.

Several noteworthy observations can be made from Fig. 6. Firstly, the original SVM features depicted in Fig. 6(a) show that the SD features and TD features are not well-separated, accounting for the poor classification performance of SVM. The performance of CORAL, shown in Fig. 6(b), is quite similar to that of SVM. However, deep structure methods such as

DCNN and DANN, whose t-SNE graphs are presented in Fig 6(c) and (d) respectively, exhibit a different behavior. They can distinguish the five different features in a high-dimensional feature space and extract representative features for distribution discrepancy matching. Despite these strengths, they struggle to maintain the differences between the individual perspective features of the SD and TD. In the case of OSBP and UDA, their t-SNE graphs, shown in Fig. 6(e) and (f), show good distribution difference alignment for common features, yielding good diagnostic performance in UUDA. Despite this, there is no significant divergence in the private features of SD and TD, which leads to some negative transfer effects due to the limitations of pseudo-label weighting and empirical selection threshold. Finally, our proposed OTDI not only addresses the distribution discrepancy between SD and TD but also presents distinguishable structures in the selected features of both domains. Notably, OTDI maintains the differences between the private features of SD and TD in the high-dimensional feature space, thus avoiding negative transfer.

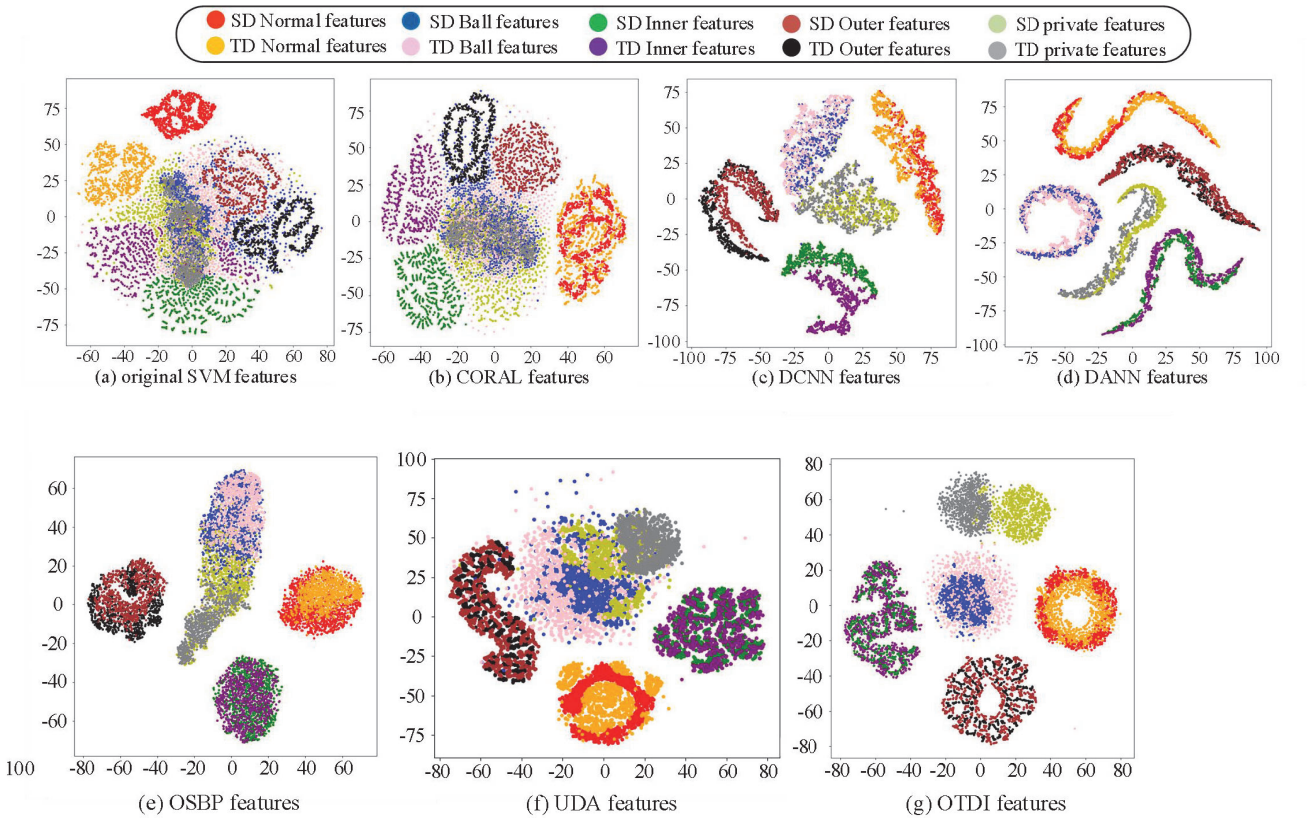


Fig.6. Features displayed by t-SNE dimensionality reduction.

4.4. Planetary Gearbox Test Results

For the planetary gearbox data, we make slight changes to some classes in the source and target domains, so that for each universal DA scenario, six subtasks are constructed separately. Table VI shows the diagnostic performance of each method in two universal DA scenarios. Again, it can be observed that the average accuracy of SVM and CORAL is far lower than other

TABLE VI

TESTING ACCURACIES OF PLANETARY GEARBOX ON DIFFERENT UNIVERSAL UDA TASKS (%)

Method	A→W						W→A						Average
	Task 1	Task 2	Task 3	Task 4	Task 5	Task 6	Task 1	Task 2	Task 3	Task 4	Task 5	Task 6	
SVM	43.25	54.94	53.28	57.07	43.25	54.94	44.78	53.87	44.78	53.87	44.08	53.54	50.14
CORAL	54.78	56.62	52.07	54.38	57.18	52.20	55.14	42. 86	41.26	47.52	43.84	44.81	50.89
DCNN	81.20	88.19	84.44	83.75	84.85	84.62	76.45	87. 96	83. 66	82.37	76.51	87.75	83.01
DANN	83.43	78.45	73.75	76.96	89.23	83.42	81.35	78. 14	82.36	78.15	83.27	70.39	80.07
OSBP	86.33	96.14	86.80	86.54	87.52	89.75	75.48	75.62	93.28	86.46	88.85	85.69	86.54
UAN	95.64	93.60	**92.61**	98.46	**97.21**	94.83	85.16	**91.75**	86.48	86.49	84.46	**94.08**	91.73
OTDI	**97.86**	**98.05**	90.99	**99.48**	96.82	**98. 90**	**92. 95**	91.03	**94.52**	**94.86**	**92.76**	91.06	**94.74**

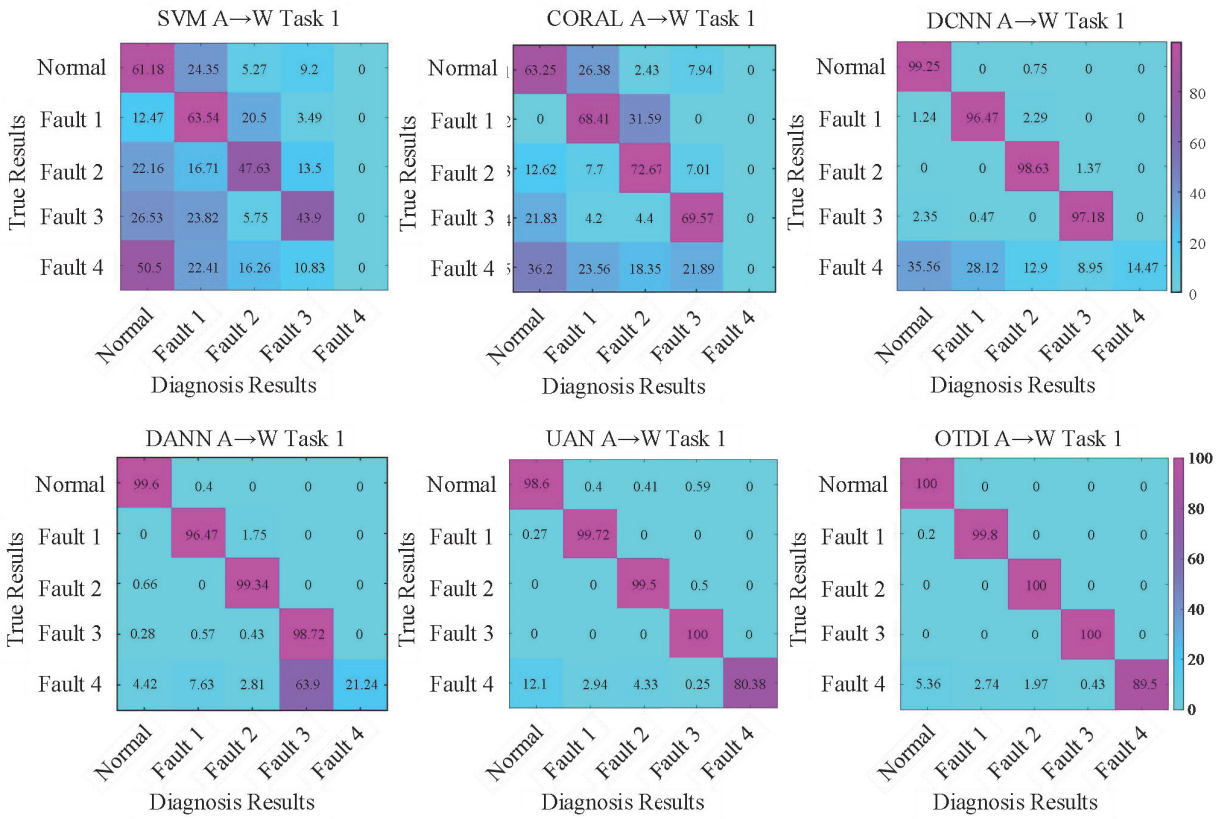


Fig.7. Confusion matrices of the six approach for gearbox in A→W Task 1.

methods. From Fig.7, we can see a similar situation in Fig.6. This can verify that supervised learning and traditional DA methods cannot solve the UUDA problem. In addition, the average accuracy of DANN is 80.07%, which is lower than that of DCNN (83.01%). This finding shows that the traditional DA can cause negative transfer in universal DA scenarios. The average accuracy rates of OSBP and UAN are 86.54% and 91.73%, respectively, which are clearly less accurate than the proposed method (OTDI) with an accuracy of 94.74%. All this confirms the effectiveness of the outlier threshold learning and domain-invariant sampler.

4.5. Ablation Study and Convergence Performance

To further evaluate the effectiveness of the designed outlier threshold learning and domain-invariant feature sampler, we have designed the following four variants of the OTDI, which are used to carry out ablation studies:

DCNN (**D**eep **C**onvolutional **N**eural **N**etwork), without using the outlier threshold and domain-invariant feature sampler. So, it is the backbone of the OTDI framework.

DCNN+BLC (**DCNN** with **b**asic **l**inear **c**lassifier). Instead of including the softmax classifier on each feature, DCNN+BLC uses a linear combination of features to make classification decisions.

DCNN+OTL (**DCNN** with **o**utlier **t**hreshold **l**earning). Instead of using an empirical value verification approach, DCNN+OTL takes into account the differences between outliers and known samples, and learns the outlier rejection thresholds through the multi-classifier.

DCNN+DFL (**DCNN** with **d**omain-invariant **f**eatures **l**earning). Instead of using a pseudo-label weighting scheme, DCNN+DFL applies the learned outlier rejection threshold to each input sample to determine whether it belongs to the common feature space.

The performance of OTDI and its variants is shown in Table IV, from which we can observe the following results: 1) OTDI consistently and significantly outperforms DCNN on the twelve UUDA tasks. This indicates that the outlier threshold learning and domain-invariant feature sampler is beneficial to the classification task in the UUDA situation. 2) OTDI shows better performance than DCNN+OTL, DCNN+DFL and DCNN+BLC on all UUDA tasks. This demonstrates that the

TABLE VII

ABLATION STUDY TESTING ACCURACIES OF ROLLING BEARING ON TWELVE UNIVERSAL UDA TASKS (%)

Method	Task 1	Task 2	Task 3	Task 4	Task 5	Task 6	Task 7	Task 8	Task 9	Task 10	Task 11	Task 12	Average
DCNN	83.14	85.92	86.3	85.74	87.62	83.76	84.86	78.74	82.44	89.14	80.86	76.16	83.72
DCNN+BLC	83.24	83.04	63.58	65.80	78.58	73.56	82.16	83.88	80.34	81.74	72.84	81.32	78.22
DCNN+OTL	91.74	92.10	82.98	80.78	87.34	91.48	89.52	85.52	87.94	85.08	76.12	83.22	85.44
DCNN+DFL	82.26	84.74	78.02	82.68	86.76	73.18	82.12	80.76	70.10	85.54	71.84	81.82	79.99
OTDI	**99.32**	**99.25**	**98.14**	**99.02**	**99.53**	**98.47**	**94.20**	**91.58**	**95.40**	**93.76**	**92.12**	**93.26**	**96.17**

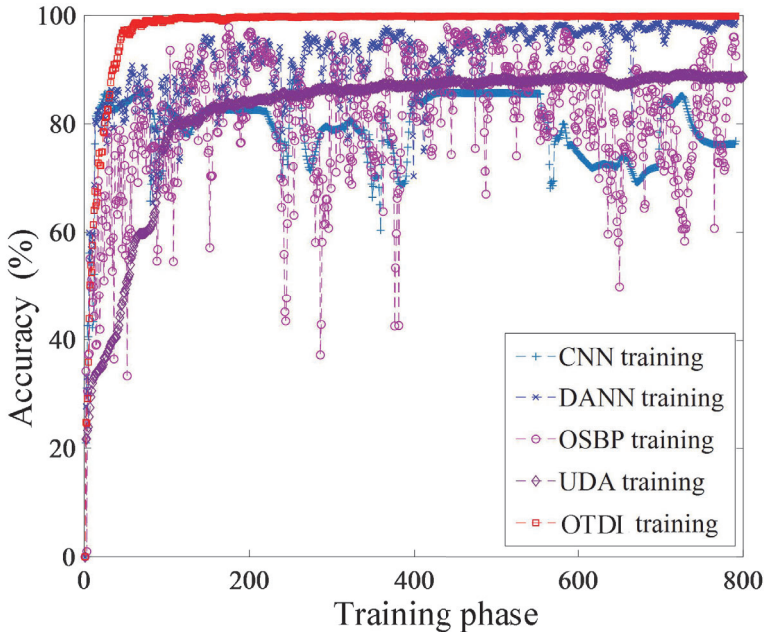


Fig.8. The convergence performance of OTDI and the compared methods against the number of iterations.

outlier threshold learning and domain-invariant feature sampler strategy is superior to the basic deep neural network architecture, pseudo-label weighting and empirical threshold value selection.

To demonstrate the robustness and effectiveness of our proposed OTDI framework, we investigated the convergence performance of the OTDI and other methods with deep structure. To save space, we only show the experimental results on the CWRU rolling bearing dataset; similar results can be obtained for the planetary gearbox dataset. We recorded the

classification accuracy of different methods (DCNN, DANN, OSBP, UDA, and OTDI) against the number of iterations under certain model training settings. Fig.6 shows the experimental results with the increasing number of iterations. We can observe that the OTDI is relatively more stable and reaches its optimal value after around 100 iterations. This shows the good robustness property of the proposed method.

5. Conclusion

In this work, we tackle the universal unsupervised domain adaptation (UUDA) by introducing an outlier threshold learning and domain-invariant features sampler for machinery fault diagnosis. Recognizing the vulnerability of empirically determined rejection thresholds to human bias, we employ an outlier threshold learning mechanism to automatically determine thresholds, which then guide the domain-invariant feature sampler. The notable advantage of our method is its data-driven nature; the outlier threshold is derived from data learning, and the domain-invariant feature sampler operates based on this threshold, thus minimizing human interference. This synergy between outlier threshold learning and the domain-invariant feature sampler strengthens our approach. The effectiveness of our proposed method has been validated through extensive experimental results on two datasets. In the future, we aim to apply the OTDI algorithm to real-world wind turbine data to assess its diagnostic performance and enhance its applicability in UUDA scenarios.

CRediT authorship contribution statement

Biliang Lu: Conceptualization, Data curation, Formal analysis, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. Yingjie Zhang: Conceptualization, Funding acquisition, Project administration, Resources, Supervision, Writing – review & editing. Zhaohua Liu: Validation, Visualization, Writing – review & editing. Hua-Liang Wei: Validation, Writing – review & editing. Qing Shuai Sun: Investigation, Writing – review & editing.

Acknowledgments

The authors gratefully acknowledge the financial support of the National Natural Science Foundation of China (No. 2019YFE0105300), the China Scholarship Council, and the Postgraduate Research & Practice Innovation Program of Hunan Province, China (QL20220095).

REFERENCES

[1] Hoang, D. T., & Kang, H. J. (2019). A survey on Deep Learning based bearing fault diagnosis. Neurocomputing, 335. https://doi.org/10.1016/j.neucom.2018.06.078

[2] Yaqub, M. F., Gondal, I., & Kamruzzaman, J. (2013). An adaptive self-configuration scheme for severity invariant machine fault diagnosis. IEEE Transactions on Reliability, 62(1). https://doi.org/10.1109/TR.2012.2222612.

[3] Zhu, Z., Cheng, J., Wang, P., Wang, J., Kang, X., & Yang, Y. (2023). A novel fault diagnosis framework for rotating machinery with hierarchical multiscale symbolic diversity entropy and robust twin hyperdisk-based tensor machine. Reliability Engineering & System Safety, 231, 109037. https://doi.org/10.1016/j.ress.2022.109037

[4] Zhu, Z., Lei, Y., Qi, G., Chai, Y., Mazur, N., An, Y., & Huang, X. (2023). A review of the application of deep learning in intelligent fault diagnosis of rotating machinery. Measurement, 206, 112346. https://doi.org/10.1016/j.measurement.2022.112346.

[5] Zhao X, Jia M, Liu Z. Semisupervised Deep Sparse Auto-Encoder With Local and Nonlocal Information for Intelligent Fault detection of Rotating Machinery. IEEE Transactions on Instrumentation and Measurement. 2021;70(3501413):1-13.https://doi: 10.1109/TIM.2020.3016045

[6] Zhao, Y., Xiao, F., & Wang, S. (2013). An intelligent chiller fault detection and diagnosis methodology using Bayesian belief network. Energy and Buildings, 57, 278-288. https://doi.org/10.1016/j.enbuild.2012.11.007.

[7] Zhao M, Zhong S, Fu X, Tang B, Pecht M. Deep Residual Shrinkage Networks for Fault detection. IEEE Transactions on Industrial Informatics. 2020;PP(99):1 – 1. https://doi. 10.1109/TII.2019.2943898

[8] Xiang, L., Zhang, X., Zhang, Y., Hu, A., & Bing, H. (2023). A novel method for rotor fault diagnosis based on deep transfer learning with simulated samples. Measurement, 207, 112350. https://doi.org/10.1016/j.measurement.2022.112350

[9] Wang, J., Zhang, Z., Liu, Z., Han, B., Bao, H., & Ji, S. (2023). Digital twin aided adversarial transfer learning method for domain adaptation fault diagnosis. Reliability Engineering & System Safety, 234, 109152. https://doi.org/10.1016/j.ress.2023.109152

[10] Yang, S., Kong, X., Wang, Q., Li, Z., Cheng, H., & Xu, K. (2022). Deep multiple auto-encoder with attention mechanism network: A dynamic domain adaptation method for rotary machine fault diagnosis under different working conditions. Knowledge-Based Systems, 249. https://doi.org/10.1016/j.knosys.2022.108639

[11] Ding, Y., Jia, M., Zhuang, J., Cao, Y., Zhao, X., & Lee, C. (2023). Deep imbalanced domain adaptation for transfer learning fault diagnosis of bearings under multiple working conditions. Reliability Engineering & System Safety, 230, 108890. https://doi.org/10.1016/j.ress.2022.108890.

[12] Shi, Y., Deng, A., Deng, M., Xu, M., Liu, Y., Ding, X., & Li, J. (2022). Transferable adaptive channel attention module for unsupervised cross-domain fault diagnosis. Reliability Engineering & System Safety, 226, 108684. https://doi.org/10.1016/j.ress.2022.108684

[13] Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., ... & Lempitsky, V. (2016). Domain-adversarial training of neural networks. The journal of machine learning research, 17(1), 2096-2030.

[14] Zhao, C., Liu, G., & Shen, W. (2022). A balanced and weighted alignment network for partial transfer fault diagnosis. ISA Transactions, 130, 449-462. https://doi.org/10.1016/j.isatra.2022.03.014

[15] Liu, X., Liu, S., Xiang, J., Sun, R., & Wei, Y. (2022). An ensemble and shared selective adversarial network for partial domain fault diagnosis of machinery. Engineering Applications of Artificial Intelligence, 113, 104906. https://doi.org/10.1016/j.engappai.2022.104906.

[16] Wei, J., Huang, H., Yao, L., Hu, Y., Fan, Q., & Huang, D. (2020). New imbalanced fault diagnosis framework based on Cluster-MWMOTE and MFO-optimized LS-SVM using limited and complex bearing data. Engineering Applications of Artificial Intelligence, 96, 103966. https://doi.org/10.1016/j.engappai.2020.103966

[17] Li, X., & Zhang, W. (2021). Deep Learning-Based Partial Domain Adaptation Method on Intelligent Machinery Fault Diagnostics. IEEE Transactions on Industrial Electronics, 68(5), 4351-4361.

[18] Deng, Y., Huang, D., Du, S., Li, G., Zhao, C., & Lv, J. (2021). A double-layer attention based adversarial network for partial transfer learning in machinery fault diagnosis. Computers in Industry, 127, 103399. https://doi.org/10.1016/j.compind.2021.103399

[19] Liang, X., Li, P., Chen, S., Jin, X., & Du, Z. (2022). Partial domain adaption based prediction calibration methodology for fault detection and diagnosis of chillers under variable operational condition scenarios. Building and Environment, 217, 109099. https://doi.org/10.1016/j.buildenv.2022.109099

[20] Saito, K., Yamamoto, S., Ushiku, Y., & Harada, T. (2018). Open Set Domain Adaptation by Backpropagation. In European Conference on Computer Vision (pp. 156-171). Springer.

[21] Lundgren, A., & Jung, D. (2022). Data-driven fault diagnosis analysis and open-set classification of time-series data. Control Engineering Practice, 121, 105006. https://doi.org/10.1016/j.conengprac.2021.105006

[22] Melo, A., Câmara, M. M., Clavijo, N., & Pinto, J. C. (2022). Open benchmarks for assessment of process monitoring and fault diagnosis techniques: A review and critical analysis. Computers & Chemical Engineering, 165, 107964. https://doi.org/10.1016/j.compchemeng.2022.107964.

[23] Zhao, C., & Shen, W. (2022). Dual adversarial network for cross-domain open set fault diagnosis. Reliability Engineering & System Safety, 221, 108358. https://doi.org/10.1016/j.ress.2022.108358.

[24] Ge, Y., Zhang, F., & Ren, Y. (2022). Adaptive fault diagnosis method for rotating machinery with unknown faults under multiple working conditions. Journal of Manufacturing Systems, 63, 177-184. https://doi.org/10.1016/j.jmsy.2022.03.009

[25] Zhang, X., He, L., Wang, X., Wang, J., & Cheng, P. (2022). Transfer fault diagnosis based on local maximum mean difference and K-means. Computers & Industrial Engineering, 172, 108568. https://doi.org/10.1016/j.cie.2022.108568

[26] Harmouche, J., Delpha, C., & Diallo, D. (2014). Incipient fault detection and diagnosis based on Kullback – Leibler divergence using Principal Component Analysis: Part I. Signal Processing, 94, 278-287. https://doi.org/10.1016/j.sigpro.2013.05.018.

[27] Chen, P., Zhao, R., He, T., Wei, K., & Yang, Q. (2022). Unsupervised domain adaptation of bearing fault diagnosis based on Join Sliced Wasserstein Distance. ISA Transactions, 129, 504-519. https://doi.org/10.1016/j.isatra.2021.12.037.

[28] Wen, L., Gao, L., & Li, X. (2019). A New Deep Transfer Learning Based on Sparse Auto-Encoder for Fault Detection. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 49(1), 136-144.

[29] Deng, Y., Huang, D., Du, S., Li, G., & Lv, J. (2021). A Double-Layer Attention Based Adversarial Network for Partial Transfer Learning in Machinery Fault Detection. Computers in Industry, 127, 103399.

[30] Zhang, W., Li, X., Ma, H., Luo, Z., & Li, X. (2021). Open-Set Domain Adaptation in Machinery Fault Diagnostics Using Instance-Level Weighted Adversarial Learning. IEEE Transactions on Industrial Informatics, PP(99), 1-1.

[31] Busto, P. P., Iqbal, A., & Gall, J. (2020). Open Set Domain Adaptation for Image and Action Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 42(2), 413-429.

[32] Loparo, K. (2013). Case Western Reserve University Bearing Data Center [Online]. Available: http://csegroups.case.edu/bearingdatacenter/pages/12k-drive-end-bearing-fault-data.

[33] Shao, S., McAleer, S., Yan, R., & Baldi, P. (2019). Highly Accurate Machine Fault detection Using Deep Transfer Learning. IEEE Transactions on Industrial Informatics, 15(4), 2446-2455. doi: 10.1109/TII.2019.2893479

[34] Zhang, J., Zhang, Q., Qin, X., & Sun, Y. (2022). A two-stage fault diagnosis methodology for rotating machinery combining optimized support vector data description and optimized support vector machine. Measurement, 200, 111651. https://doi.org/10.1016/j.measurement.2022.111651.

[35] Zhang, J., Kong, x., Li, x., Hu, z., Cheng, l., & Yu, m. (2022). Fault diagnosis of bearings based on deep separable convolutional neural network and spatial dropout. Chinese Journal of Aeronautics, 35(10), 301-312. https://doi.org/10.1016/j.cja.2022.03.007.

[36] Li, X., Jiang, H., Liu, S., Zhang, J., & Xu, J. (2021). A unified framework incorporating predictive generative denoising autoencoder and deep Coral network for rolling bearing fault diagnosis with unbalanced data. Measurement, 178, 109345. https://doi.org/10.1016/j.measurement.2021.109345.

[37] Yu, X., Zhao, Z., Zhang, X., Chen, X., & Cai, J. (2023). Statistical identification guided open-set domain adaptation in fault diagnosis. Reliability Engineering & System Safety, 232, 109047. https://doi.org/10.1016/j.ress.2022.109047.

[38] Zhang, Y., Ren, Z., Feng, K., Yu, K., Beer, M., & Liu, Z. (2023). Universal source-free domain adaptation method for cross-domain fault diagnosis of machines. Mechanical Systems and Signal Processing, 191, 110159. https://doi.org/10.1016/j.ymssp.2023.110159.

[39] Melit Devassy, B., & George, S. (2020). Dimensionality reduction and visualisation of hyperspectral ink data using t-SNE. Forensic Science International, 311, 110194. https://doi.org/10.1016/j.forsciint.2020.110194.

Declaration of interests

☒The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: