Journal of Intelligent Manufacturing (2022) 33:973–983
https://doi.org/10.1007/s10845-020-01709-4

Deep prototypical networks based domain adaptation for fault diagnosis

Huanjie Wang[1,2] · Xiwei Bai[1,2] · Jie Tan[1] · Jiechao Yang[1,2]

Received: 13 March 2020 / Accepted: 29 October 2020 / Published online: 11 November 2020

Abstract

Due to the existence of domain shifts, the distributions of data acquired from different machines show significant discrepancies in industrial applications, which leads to performance degradation of traditional machine learning methods. In this paper, we propose a novel method that combines supervised domain adaptation with prototype learning for fault diagnosis. The proposed method consists of two modules, i.e., feature learning and condition recognition. The module of feature learning applies the Siamese architecture based on one-dimensional convolutional neural networks to learn a domain invariant subspace, which reduces the inter-domain discrepancy of distributions. The module of condition recognition applies a prototypical layer to learn the prototypes of each class. Then the classification task is simplified to find the nearest class prototype. Compared with existing intelligent fault diagnosis methods, this proposed method can be extended to address the problem when the classes from the source and target domains are partially overlapped. The model must generalize to unknown classes in the source domain, given only a few samples of each new target class. The effectiveness of the proposed method is verified using two bearing datasets. The model quickly converges with high classification accuracy using a few labeled target samples in training, even one per class can be effective.

Keywords Bearing · Fault diagnosis · Domain adaptation · Prototype learning

Introduction

Rolling element bearings are precision components in rotating machines, which are widely used in industrial, automotive, aerospace and marine applications (Ai 2013). With the development of advanced manufacturing technology, various sensors (e.g., temperature, vibration, displacement) have been utilized to monitor the condition of the bearing. The vibration signals that contain machine health information have proven to be effective for fault diagnosis and prognosis of bearings. This has promoted a great deal of work on vibration analysis over the last few decades (Yu et al. 2006; Sreejith et al. 2008; Wen et al. 2017b; Gao et al. 2019; Chen et al. 2020). Yu et al. (2006) proposed a method based on empirical mode decomposition (EMD) energy entropy for bearing fault diagnosis. EMD is applied to decompose the raw vibration signals and obtain intrinsic mode functions (IMFs). Then the extracted energy features of the IMFs are taken as input to the artificial neural network to distinguish normal bearing. The method proposed by Sreejith et al. (2008) extracts the Normal negative log-likelihood value and kurtosis value from time-domain vibration signals and uses these values as the input to the neural network. The proposed method can distinguish bearing conditions with high accuracy. Wen et al. (2017b) proposed a signal-to-image conversion method. The proposed method converts the raw vibration signals into two-dimensional gray images. Then a convolutional neural network (CNN) is applied to extract the features of these two-dimensional gray images, which can eliminate the effect of handcrafted features.

Despite the impressive performance of the above methods, the data-driven methods require a large amount of labeled data (target data) in training, which restricts their extensive applications. In industrial applications, the machine working conditions are generally complicated and frequently change over time. It is difficult to collect and label sufficient samples for various types of faults under different working condi-

✉ Jie Tan
jie.tan@ia.ac.cn

[1] Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China

[2] School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China

tions. Meanwhile, the fault detection system does not allow critical machines to operate in fault states. Once the system detects a fault condition, it will immediately shut down the machine, which results in collecting only a few fault samples. Although it is difficult to obtain massive labeled data for critical machines, it can obtain enough data from different but related machines. However, the data collected from different conditions show significant distribution discrepancies. It indicates that the model trained in one situation is not suitable for another. It is difficult or even impossible to recollect the new labeled data to train a model for the actual task. When it is difficult to collect the target data, the typical method is to train the related target model using available data (source data). Since this method may lead to suboptimal performance, techniques such as transfer learning (Wen et al. 2017a; Han et al. 2020), generative model (Li et al. 2018a; Liu et al. 2018), and few-shot learning (Zhang et al. 2019) have been studied to address the issue for fault diagnosis.

Domain adaptation, which can transfer the knowledge from the source domain to a different but related target domain, can be adopted in the situation where the source and target data have different distributions. Although the methods based on domain adaptation have been widely used in situations such as image classification, face recognition, object detection, and so forth (Wang and Deng 2018), they have not yet been investigated thoroughly for fault diagnosis. In recent years, several fault diagnosis methods based on domain adaptation have been proposed (Lu et al. 2016; Li et al. 2018b; Tong et al. 2018; Li et al. 2019). These methods effectively transfer useful information from the labeled source domain to the unlabeled target domain. However, there are rare effective fault data in real-world applications. It is hard to collect massive unlabeled target samples covering various types of faults, which indicates the target dataset is generally imbalanced. Moreover, most unsupervised transfer learning methods use distance metrics such as Maximum Mean Discrepancy (MMD) (Gretton et al. 2007) to measure the discrepancy between different domains. These methods mainly focus on the transferability between different working conditions. Generally, these methods cannot effectively distinguish the fault types collected from different machines. Then data-driven fault diagnosis methods centered on transferability between machines are attracting ever-increasing attention (Guo et al. 2018; Zhao et al. 2020).

In this work, we propose a supervised domain adaptation method for bearing fault diagnosis. The main idea of the method is to apply the Siamese architecture to learn a domain invariant subspace, followed by a prototypical layer that computes the prototypes of each class. The modified model is based on the assumption that there exist domain invariant prototypes that can represent corresponding classes in the latent space. To do this, we learn a non-linear mapping to minimize the discrepancy between source and target distributions in a latent space and take a prototype to represent each class. Then the classification task is simplified to find the nearest class prototype. The method requires only a few labeled target samples in training. Even one sample per class can significantly improve model performance. These labeled target samples are used as prior knowledge to determine the similarity between the different domains. Furthermore, the model trained on the source domain can generalize to new classes that can only be seen in the target domain. With the domain invariant prototypes, the model uses the distance from extract features to the closest class center as an outlier score to distinguish the unknown classes. The proposed method is verified on two famous bearing datasets, which shows that our method is effective in fault diagnosis with a few labeled target samples. The main contributions of this work are summarized in the following.

1. The proposed method uses convolutional neural networks as a basic model applied to fault diagnosis. It can learn a domain invariant subspace with effective transferability with the Siamese architecture.
2. The proposed method can be extended to solve the problem when the classes from different domains are partially overlapped. The model must generalize to unknown classes in the source domain.
3. With only a few labeled target samples in training, the model can distinguish target classes and reject the samples from unknown source classes.
4. With attractive robustness, randomly assigning corresponding labels between source and target domains does not affect the model performance, which is suitable for the complex working conditions in industrial applications.

The rest of the paper is organized as follows. “Related works” section reviews related works about domain adaptation and prototype learning. “Proposed method” section describes the problem formulation and the proposed method. A series of experiments are carried out in “Experimental results” section. Finally, conclusions and future works are presented in “Conclusion” section.

Related works

Domain adaptation (DA) has attracted ever-increasing attention for reducing the annotation burdens in the target domain. As mentioned previously, most unsupervised DA-based methods focus on the situation where different domains are completely overlapped. These methods assume that the source domain contains the same tasks as the target domain. The base model takes a batch of labeled source samples and unlabeled target samples and directly minimizes the discrepancy between their distributions. This procedure aligns all

target samples with source classes, which makes it difficult to distinguish unknown classes in training. Moreover, such methods usually do not perform well in the DA settings where a few labeled target samples are available (Saito et al. 2019). It is difficult to obtain the discriminative features with the limited target samples. Therefore, here are some supervised DA-based methods that can overcome the limitations of unsupervised DA-based methods and improve the classification accuracy of the target domain with a few labeled samples per class. Hoffman et al. (2013) proposed an iterative process that simultaneously learns the classifier weights and a transformation to map target features to the source domain. Motiian et al. (2017) proposed a weakly-supervised framework applied to visual domain adaptation and generalization. The method utilizes the Siamese architecture to learn a discriminative embedding subspace, where the mapped features are inter-class separable and intra-class similar for both domains. Saito et al. (2019) proposed a novel adversarial method, Minimax Entropy (MME), to extract discriminative features. The method regards the class weight vectors as estimated prototypes and minimizes the distance between the prototypes and corresponding unlabeled target samples. These methods can take advantage of a few knowing target samples with supervised DA. Since the new target samples are severely limited for the above problem, the traditional classification layers are challenging to separate the new classes from each other. Therefore, we modify the model by prototype learning, which improves model performance to extract discriminative features. Then the model can be extended to reject the unknown source classes as outliers.

Through searching prototypes to represent the distribution of each class, prototype learning is effective in improving the performance of classification. The simplest method of prototype learning is the unsupervised clustering, which searches the class centers used as the reduced prototypes independently (Bezdek et al. 1998; Liu and Nakagawa 2001). Since the unsupervised clustering does not consider the class information, the classification accuracy is generally lower compared with supervised classification methods. The learning vector quantization (LVQ) (Kohonen 1990), proposed by Kohonen, supervised adjusts the weight vectors based on searching the optimal position of the prototypes. Although the convergence is not guaranteed, the attractive performance makes LVQ popular in many works. In the variations of LVQ, the parameter optimization methods, which learn prototypes through optimizing the objective functions by gradient search, have excellent convergence property in learning (Sato and Yamada 1996, 1998). Snell et al. (2017) proposed prototypical networks for both few-shot and zero-shot classification. The method learns prototype representations of each class in a metric space by computing the mean of embedded support examples. Yang et al. (2018) combined the deep CNN with the prototype-based classifier, which improved the model robustness. In this work, the labeled target samples are used as prior knowledge to learn the domain invariant subspace. To test the robustness of the proposed method, we randomly assign class labels between different domains in the experiments.

Table 1 Notations and descriptions

Notation	Description	Notation	Description
$\mathcal{D}$	Domain	s, t	Source, target
X	Data set	Y	Label set
x	Single sample	y	Corresponding label
$\mathcal{X}$	Data space	$\mathcal{Y}$	Label space
$\mathcal{Z}$	Latent space	Z	Feature set
M	Class number	m	Sample size
c	Class prototype	C	Prototype set
P	Data distribution	f	Deep model
h	Feature learning	g	Condition recognition

Proposed method

Notation

In this section, we describe the problem formulation and the proposed method to address supervised DA. Let $\mathcal{D}_s = \{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^{m_s}$ denotes the source dataset $\mathcal{D}_s$ that consists of m_s data points. Here, $\mathbf{x}_i^s \in \mathbb{R}^D$ represents the D-dimensional input sample, and $y_i^s \in \{1, \ldots, M^s\}$ is the corresponding label. A target domain given a limited number of labeled samples is denoted by $\mathcal{D}_t = \{(\mathbf{x}_i^t, y_i^t)\}_{i=1}^{m_t}$. X^s, X^t denote the sets of $\mathbf{x}_i^s, \mathbf{x}_i^t$ respectively, and $y_i^t \in \{1, \ldots, M^t\}$. The Table 1 describes the notations which are used in this work frequently.

DA-based methods generally assume there is a covariate shift (Shimodaira 2000) between X^s and X^t, i.e., $P(X^s) \neq P(X^t)$. The performance of the model trained in the source domain may drop dramatically when it is applied in the target domain. In this work, we aim to learn a deep model $f : \mathcal{X} \rightarrow \mathcal{Y}$ that can work well on both source and target domains. The model f is composed of two components, i.e., $f(\mathbf{x}_i, \theta) = g(h(\mathbf{x}_i, \theta_1), \theta_2)$. Here $h : \mathcal{X} \rightarrow \mathcal{Z}$, the feature learning module, is a mapping from the data space $\mathcal{X}$ to a latent space $\mathcal{Z}$, and $g : \mathcal{Z} \rightarrow \mathcal{Y}$ is a condition recognition module to predict the corresponding label. The $\theta = \{\theta_1, \theta_2\}$ denotes the parameters of the model. To simplify the model, we only learn one prototype for each class in this work. The prototype is an N-dimensional vector denoted as $\mathbf{c}_i \in \mathbb{R}^N$, and $i \in \{1, 2, \ldots, M^t\}$ represents the index of the predicted class.

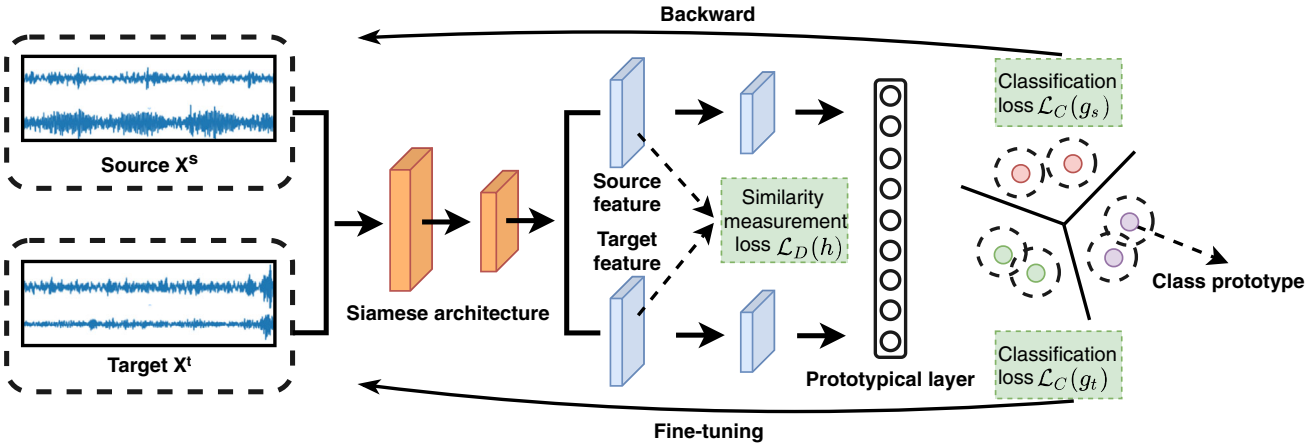

Fig. 1 Deep prototypical networks based DA (DPDAN): Two input points are multi-dimensional vibration signals selected from the source and target domains, respectively. During the training stage, the source training samples contain the labeled target samples. The distance loss $\mathcal{L}_D$ minimizes the distribution discrepancy between different domains to learn a domain invariant subspace, and the classification loss makes features discriminative to learn the prototypes. In this figure, the model learns two prototypes for each class and utilizes prototype matching for classification. The prototype matching method aligns the samples with the nearest prototype

Architecture

As shown in Fig. 1, the proposed model consists of two parts: a feature learning module h and a condition recognition module g. The h is achieved by the Siamese architecture based on one-dimensional convolutional neural networks. The convolutional layers perform some non-linearities to convert the multi-dimensional raw vibration data into abstract features. To avoid the interference of the high-frequency noises in industrial environments, wide first-layer kernels (Zhang et al. 2017) are used to extract features, followed by small kernels to get discriminative feature representations. Then dropout (Hinton et al. 2012) is applied before the output of the h to avoid overfitting. With the dropout, binary variables $\text{diag}(\epsilon)$ are sampled with a fixed probability p, i.e., $\epsilon \sim \text{Bernoulli}(p)$. When the sampled binary variable is value 0, the corresponding unit is dropped in training. Subsequently, the g takes the prototypical layer to transform abstract features into N-dimensional vectors. Then the distances between the N-dimensional vectors and the learned prototypes are used to estimate conditions of the tested bearing. Compared with traditional classification layers, the extracted features can make a certain degree of change around the corresponding prototypes, which improves the generalization performance.

Feature learning

With the covariate shift assumption of DA, we can assume that the source and target domain have the same conditional probability distribution, i.e., $P(Y^s|X^s) = P(Y^t|X^t)$. It means that the condition recognition modules of the source and target domains could be the same when we learn a domain invariant space for their distributions, i.e., $g_s = g_t$. Meanwhile, the parameters of CNN can be shared in a Siamese architecture, i.e., $h_s = h_t$. Therefore, there exists a model that can work well on both source and target domains. The h computes the pairwise distances in the latent space to minimize the distribution discrepancy in training. In that case, the method assumes that $h_s = h_t$, and we could train the function h by minimizing a distance loss

$$\mathcal{L}_D(h, \theta_1) = E\left[\ell\left(d\left(h\left(X^s\right), h\left(X^t\right)\right), Y_d\right)\right] \tag{1}$$

where the $E[\cdot]$ denotes the mathematical expectation and the $y_d \in \{0, 1\}$ denotes the consistency of two input points. When the two input points $(\mathbf{x}_i^s, \mathbf{x}_i^t)$ are sampled from the same class, $y_d = 1$, otherwise, $y_d = 0$. Binary cross-entropy loss is used as ℓ in this work, and the function d computes the pairwise distances between features in the latent space. In this work, the function d is computed with Euclidean distance. Then their consistency is estimated by

$$d\left(h\left(X^s\right), h\left(X^t\right)\right) = \frac{1}{1 + e^{-\|Z^s - Z^t\|_2/T}} = \mathcal{S}(\boldsymbol{\eta}/T) \tag{2}$$

where $\boldsymbol{\eta} = \left\|Z^s - Z^t\right\|_2$, and $\mathcal{S}$ denotes the Sigmoid function. Then $\boldsymbol{\eta}$ is divided by a constant T called the temperature that controls the distance mapping, and T is set to -0.5 in this work. The CNN architecture is detailed in Table 2.

Condition recognition

With features obtained from h, most CNN-based methods usually use the softmax layer for classification. Compared

Table 2 Structure of the feature learning module

Layer	Name	Size/Stride
1	Convolutional-ReLU	16 filters of $64 \times 1/1 \times 1$
2	Max-Pooling	$2 \times 1/2 \times 1$
3	Convolutional-ReLU	32 filters of $3 \times 1/1 \times 1$
4	Max-Pooling	$2 \times 1/2 \times 1$
5	Convolutional-ReLU	64 filters of $2 \times 1/1 \times 1$
6	Max-Pooling	$2 \times 1/2 \times 1$
7	Convolutional-ReLU	64 filters of $3 \times 1/1 \times 1$
8	Max-Pooling	$2 \times 1/2 \times 1$
9	Convolutional-ReLU	64 filters of $3 \times 1/1 \times 1$
10	Max-Pooling	$2 \times 1/2 \times 1$
11	Fully-connected layer	100

with the conventional softmax layer, the prototypical layer projects the samples around the learned prototypes in the latent space, which leaves large regions for unknown classes. To simplify the model, we only learn one prototype for each class, and the prototypical layer outputs an N-dimensional vector for each class to approximate the corresponding prototype. To learn the prototypes, a distance metric (e.g., Euclidean distance) is used to compute the similarity between the N-dimensional vectors and the prototypes. The model has two parts of trainable parameters, one for the model f and the other for the prototypes C. Given the feature set Z obtained from h, we further define the classification loss as

$$\mathcal{L}_C(g, \theta_2) = E\left[\ell\left(P\left(C|Z, \theta_2\right), Y\right)\right] + \lambda_c \|g(Z) - C_m\|_1 \quad (3)$$

where ℓ denotes categorical cross-entropy that controls the classification accuracy. C_m denotes the corresponding prototype of input X. λ_c is the hyper-parameter that changes the weight of the regularization and is set to 0.5 in this work. For the given class i, the probability $P\left(C_i|Z, \theta_2\right)$ is defined as

$$P\left(C_i|Z, \theta_2\right) = \frac{e^{-\gamma d(g(Z), C_i)}}{\sum_{l=1}^{M^t} e^{-\gamma d(g(Z), C_l)}} \quad (4)$$

where γ is a hyper-parameter that controls assignments of the distance to probability. The distance is measured by the function, $d(g(Z), C_i) = \|g(Z) - C_i\|_2^2$. By limiting the distance between learned vectors and the corresponding prototypes, the regularization $\|g(Z) - C_m\|_1$ makes the features in the same classes more compact. Finally, we get the modified method by learning a deep model f such that

$$\mathcal{L}_P(f, \theta) = \lambda \mathcal{L}_D(h, \theta_1) + (1 - \lambda)\mathcal{L}_C(g, \theta_2) \quad (5)$$

Based on the assumption that $g_s = g_t = g$, the prototypical layer g is trained with source data, and then fine-tuned based on a few labeled target samples.

$$\begin{aligned} g_s &= \text{train}(g|\mathcal{D}_s) \\ g_t &= \text{fine-tune}(g|\mathcal{D}_t) \end{aligned} \quad (6)$$

Application

Given the model f that can distinguish the classes provided in training, the second issue is to effectively recognize the unknown classes in the source domain. The graphic description of the procedure is shown in Fig. 2. Compared with other CNN-based methods, the model f projects the samples to some specific regions of the latent space (around the prototypes), which leaves large regions for unknown classes. Then we apply the model f to obtain the outlier score that indicates the degree to which the f estimates a sample $\mathbf{x}_i$ to be an outlier. The outlier score represents the distance from $\mathbf{z}_i$ to the closest class center. It is calculated as

$$\text{score}(\mathbf{x}_i) = \min_{1 \le j \le M^s} \left\|\boldsymbol{\mu}_j - \mathbf{z}_i\right\|_2^2 \quad (7)$$

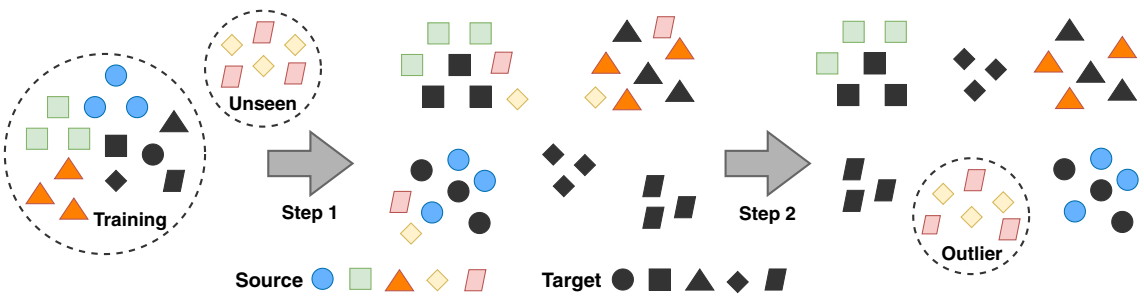

Fig. 2 Overview of the proposed method. We assume the classes from different domains are partially overlapped. There are some unknown source classes in training. In the first step, we train the model f to recognize the known classes in training. Then we utilize discriminative features obtained from the h to learn the outlier scores which are used to recognize the unknown source classes

Table 3 Description of the datasets provided by the CWRU

Diameter (in.)	NC	BF			IF			OF (6:00)			Speed	Dataset	Size	Position
	0	0.007	0.014	0.021	0.007	0.014	0.021	0.007	0.014	0.021	(rpm)			
Labels	1	2	3	4	5	6	7	8	9	10	1730	$\mathcal{D}_A$	1250×10	Drive-end
Labels	1	2	3	4	5	6	7	8	9	10	1797	$\mathcal{D}_B$	1250×10	Drive-end
Labels	1	2	3	4	5	6	7	8	9	10	1797	$\mathcal{D}_C$	1250×10	Fan-end
Labels	1	2	3	*	5	*	*	8	*	10	1730	$\mathcal{D}_D$	1250×6	Drive-end
Labels	1	2	3	*	5	*	*	8	*	10	1797	$\mathcal{D}_E$	1250×6	Fan-end
Labels	1	10	7	4	3	6	8	9	5	2	1797	$\mathcal{D}_F$	1250×10	Fan-end

where $\mathbf{z}_i = h(\mathbf{x}_i, \theta_1)$. $\boldsymbol{\mu}_j$ is the mean of the class j and calculated as

$$\boldsymbol{\mu}_j = \frac{1}{|X_j^s|} \sum_{i=1}^{|X_j^s|} \mathbf{z}_i^s \tag{8}$$

Then we adopt the threshold-based rejection strategy with the outlier scores. The threshold value determines the distance for outlier detection. If the score for a sample is larger than the estimated threshold, the sample will be rejected. For the threshold values, we adopt the threshold estimation method (Hassen and Chan 2020) which assumes a certain percentage of the training samples are outliers in this work. We calculate the outlier scores on the training data of the source domain. Then the outlier scores are sorted in ascending order, and the 99 percentile outlier score in each class is used as the threshold value.

Experimental results

In this section, the proposed CNN-based method is conducted on two fault diagnosis datasets, that is, the Case Western Reserve University (CWRU) bearing dataset (Smith and Randall 2015) and Intelligent Maintenance Systems (IMS) bearing dataset (Qiu et al. 2006).

1. CWRU Bearing Dataset: CWRU bearing dataset was collected from a bearing test stand provided by the CWRU. The vibration data used in this paper were collected from different ends of the motor (drive-end and fan-end) under three load scenarios (0, 2, 3 horsepower) and on four different health conditions, i.e., normal condition (NC), ball fault (BF), inner race fault (IF) and outer race fault (OF).
2. IMS Bearing Dataset: IMS bearing dataset was provided by the Center for IMS, University of Cincinnati. A radial load of 6000 lbs was applied onto the tested bearings in test-to-failure experiments. At the end of the experiments, BF, IF and OF occurred in three bearings, respectively. In this paper, the data collected from three fault conditions and one normal condition are used to construct a bearing dataset.

The first three experiments are conducted on the CWRU bearing data collected from different ends. We select the fault diameters of 0.007, 0.014, and 0.021 inches for every type of fault and have ten conditions in total added with a normal condition. The samples are generated by the sliding window of 2048 size with 80 points shift step. As shown in Table 3, these datasets contain 1250 samples per class, and $\mathcal{D}_D$ and $\mathcal{D}_E$ have only six classes. Then the proposed method is further evaluated on the fourth transfer task between CWRU bearing dataset and IMS bearing dataset. As shown in Table 7, there are four different health conditions in these datasets, and each contains 500 samples per class. Similarly, these samples are generated by the sliding window of 1200 size with 100 points shift step.

To verify the effectiveness of the proposed DPDAN model, several methods are used for comparison on the experiments.

1. support vector machines (SVM) (Hsu and Lin 2002);
2. base CNN without the DA technique;
3. transfer component analysis (TCA) (Pan et al. 2010);
4. deep domain confusion (DDC) (Tzeng et al. 2014);
5. our proposed DPDAN model;
6. the DSDAN model which replaces the prototypical layer with the conventional softmax layer.

The first type of comparison methods is used to evaluate the improvement of DA-based methods for fault diagnosis. First, SVM and base CNN are trained only using the source data. Then SVM is trained using the features obtained from h to evaluate the effects of the features learned by DPDAN. TCA is a popular DA-based method that learns a shared subspace between different domains. The distributions of different domains are close to each other in the shared subspace. Two features, (handcrafted features, automatic features learned by CNN) are used to train the model in this paper. DDC is a deep DA-based method for image classification prob-

Table 4 Results (%) of the completely overlapped setting

Method	A→B	B→A	A→C	C→A	Average
SVM	42.6	36.3	20.3	22.4	30.4
SVM-*h*	99.2	100	99.3	96.7	98.8
TCA	76.6	74.8	26.3	26.6	51.1
TCA-CNN	91.2	90.1	68.8	62.7	78.2
CNN	67.8	62.9	21.6	18.1	42.6
DDC	77.6	78.1	31.2	28.7	53.9
DSDAN-1	98.7	100	98.1	99.1	99.0
DSDAN-3	99.5	100	99.5	100	99.8
DPDAN-1	98.8	100	98.3	98.6	98.9
DPDAN-3	99.2	100	99.4	100	99.7

The method-*n* stands for the method using *n* labeled target samples per class in training. The method-*h* stands for the method using features obtained from *h*

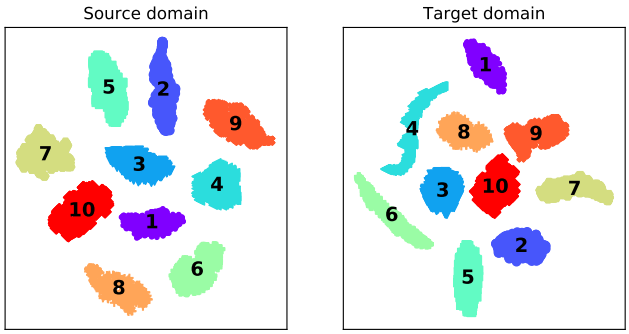

Fig. 3 *t-SNE* visualization of features: 100-dimensional feature vectors obtained from *h* are reduced into a two-dimensional map. The class-*i* ($i \in \{4, 6, 7, 9\}$) data of the source domain are unknown in training

lems. The method applies an adaptation layer to minimize the MMD-based distances between different domains. The domain confusion loss is trained to optimize for domain invariance. DDC is used to evaluate the robustness of our proposed method. During the training stage, the proposed method is initialized by the weight initialization proposed by He et al. (2015). Then the Adam algorithm (Kingma and Ba 2014) with minibatch stochastic gradient descent is used to optimize the model parameters. The batch size and initial learning rate are set to 64 and 0.001, respectively. We randomly choose n samples per class in the target domain for five times to generate different training sets and calculate the mean of accuracies.

Completely overlapped domain adaptation

As shown in Table 3, we evaluate the proposed method on three datasets ($\mathcal{D}_A$, $\mathcal{D}_B$, and $\mathcal{D}_C$) that were generated from CWRU bearing dataset. The data of $\mathcal{D}_A$ and $\mathcal{D}_B$ are collected from the drive-end of the test stand, which is different from $\mathcal{D}_C$. First, n ($n \in \{1, 3\}$) samples per class in the target dataset are randomly selected as training data. The rest of the target data are used for testing. Then the proposed model is trained using the source dataset and the selected target samples.

Table 4 reports the classification accuracies of six methods. For TCA, 6 handcrafted features, i.e., root mean square (RMS), variance, kurtosis, skewness, crest factor, approximate entropy are extracted to train the model. The subspace dimension of TCA is set to 8 in this paper. Then an SVM classifier is trained on the reduced dimensional features of the source data to classify the unlabeled target data. From the results, it shows the effects of distribution discrepancy on model performance. The data collected from the same end have more similar distributions and obtain higher classification accuracies. For TCA-CNN, a one-dimensional CNN is trained using the labeled data in the source and target domains. The trained CNN is used to extract the features from both domains. Then TCA uses the extracted features to estimate the conditions of the tested bearing. The results of TCA-CNN show the effects of labeled target samples used in training. To evaluate the effects of learned features of our proposed methods, SVM uses features obtained from h to estimate conditions of the tested bearing. With the features learned by DPDAN-3, the performance of the SVM-*h* has been greatly improved.

To fairly evaluate the performance, the architectures of different deep models are built as similar as possible. DDC adopts an adaptation layer and minimizes the MMD distance between the source and target domains. Since only a single layer of the network is adapted, the architecture of DDC (e.g., the position and width of the adaptation layer) restricts the model performance. Moreover, DDC is sensitive to the distributions of different domains due to the limitation of prior knowledge. Compared with these methods, our proposed method performs well on different tasks, even only ten labeled target samples ($n = 1$, one sample per class) are used in training.

Partially overlapped domain adaptation

Existing fault diagnosis methods based on DA generally assume the source and target domains contain the identical label space. However, it is difficult to obtain data covering various types of faults for critical machines. As shown in Table 5, the proposed method is conducted on two transfer tasks, i.e., $\mathcal{D}_D \rightarrow \mathcal{D}_C$ and $\mathcal{D}_E \rightarrow \mathcal{D}_A$. Compared with completely overlapped DA, the proposed model is applied to accommodate ten classes in the target domain when the source domain contains six classes in training. Table 5 shows the classification accuracies increase with the size of labeled target samples ($n \in \{1, 3, 5\}$) rising. To demonstrate the performance directly, we follow the *t-SNE* (Maaten and Hinton 2008) to visualize the high dimensional features obtained

Table 5 Results (%) of the partially overlapped setting

Method	D→C	E→A	Average
SVM	18.3	30.1	24.2
CNN	17.9	31.3	24.6
DSDAN-1	63.5	65.7	64.6
DSDAN-3	85.4	88.7	87.1
DSDAN-5	96.2	98.6	97.4
DPDAN-1	62.3	68.2	65.3
DPDAN-3	91.3	96.3	93.8
DPDAN-5	99.4	99.6	99.5

The method-n stands for the method using n labeled target samples per class in training

Fig. 4 Average classification accuracy for the partially overlapped tasks with different numbers of known classes in the source domain. The red lines represent the $\mathcal{D}_D \rightarrow \mathcal{D}_C$ task, and the green lines represent the $\mathcal{D}_E \rightarrow \mathcal{D}_A$ task

from h into a two-dimensional map. From Fig. 3, we can find that the proposed method learns the latent space where the features of both domains are inter-class separable.

Moreover, the features of unknown classes in the source domain are discriminative. We further extend the model to recognize the unknown classes in the source domain. To report the recognition ability of the unknown classes, six known-class data are fed to the DPDAN model and calculated the outlier scores. Then the 99 percentile outlier score in each class is used as the threshold value. We use a measurement of rejection rate (RR), which denotes the percentage of rejected samples of the unknown classes, to evaluate the rejection ability. By comparing with the pre-defined threshold, over 95% of unknown-class samples can be rejected.

To evaluate the influence of the overlapping classes in training, five partial overlapping situations are conducted in this paper. As shown in Fig. 4, the proposed model can effectively recognize the corresponding target classes of source known classes with only one sample. Moreover, the model quickly converges with high classification accuracy in the target domain with the size of labeled target samples rising.

Table 6 Results (%) of randomized label assignments

Method	A→F	F→A	Average
SVM	17.9	10.2	14.1
SVM-h	99.1	99.5	99.3
TCA	23.0	25.2	24.1
TCA-CNN	64.7	57.3	61.0
CNN	21.0	26.6	23.8
DDC	21.4	16.2	18.8
DSDAN-1	98.4	98.5	98.5
DSDAN-3	99.6	99.8	99.7
DPDAN-1	98.0	98.4	98.2
DPDAN-3	99.3	99.6	99.5

The method-n stands for the method using n labeled target samples per class in training. The method-h stands for the method using features obtained from h

Table 7 Description of bearing datasets

Dataset	Bearing	Condition	Speed (rpm)	Size
$\mathcal{D}_U$	CWRU	NC	1750	500×4
		BF	1750	
		IF	1750	
		OF	1750	
$\mathcal{D}_S$	IMS	NC	2000	500×4
		BF	2000	
		IF	2000	
		OF	2000	

Randomized label assignments

In industrial applications, we may meet the situation where no labeled data are available in the target domain. To address the fault diagnosis problem with unlabeled data, we select the representations of different types of faults by evaluating the difference between the unlabeled data. Since we do not know the types of faults in advance, the representative sample labels will be randomly assigned. To verify the robustness of the proposed method, we randomly assigned the class labels of $\mathcal{D}_C$ to get $\mathcal{D}_F$. In the third experiment, we follow the setting of the first experiment but replaced $\mathcal{D}_C$ with $\mathcal{D}_F$. From Table 6, we can see that randomized label assignments do not cause a significant drop in model performance. The labeled target samples are used as prior knowledge to determine the transferable features between the source and target domains. Then some specific regions of the latent space are learned on the extracted features, which can compensate for great domain shifts.

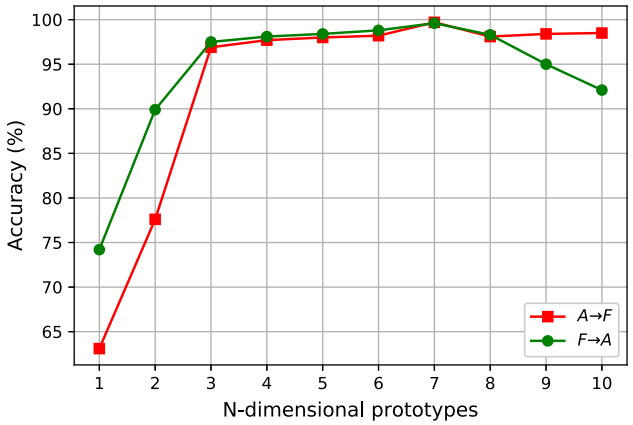

Fig. 5 Average classification accuracy for randomized label assignments using DPDAN-1

Table 8 Results (%) of validation experiments

Method	U→S	S→U	Average
SVM	24.8	25.0	24.9
SVM-*h*	98.7	98.3	98.5
TCA	16.0	10.2	13.1
TCA-CNN	36.9	38.7	37.8
CNN	24.9	42.1	33.5
DDC	48.5	49.8	49.2
DCTLN	89.7	89.9	89.8
DSDAN-1	74.5	79.1	76.8
DSDAN-3	98.1	93.2	95.7
DPDAN-1	82.4	87.6	85.0
DPDAN-3	98.6	98.1	98.4

The method-*n* stands for the method using *n* labeled target samples per class in training. The method-*h* stands for the method using features obtained from *h*

In the previous experiments, the dimension of the learned prototypes is set to 5. To evaluate the influence of dimension N, we adopt different values ($N \in \{1, \ldots, 10\}$) for the prototypes. For different N, the same settings of the hyperparameters are used during the training, and the results are shown in Fig. 5. From the results, it can be seen that the performance of the method is improved with the increase of N within a certain range. Then the performance remains unchanged or even decreased as the N increases in training.

Validation on IMS bearing dataset

To further validate the proposed method, two datasets generated from CWRU bearing dataset and IMS bearing dataset are used to conduct transfer experiments. We follow a similar experiment setting of the above experiments. In each experiment, the training dataset contains all the labeled source data and a few labeled target data. The rest of the target data are used for testing. The generated datasets are composed of one-dimensional vibration signals, which is different from previous experiments. Therefore, the proposed model is simplified for the new transfer task. The results of the two transfer experiments are shown in Table 8. DCTLN is an unsupervised DA-based method proposed by Guo et al. (2018). The method consists of two modules, i.e., condition recognition and domain adaptation. The condition recognition module is used to automatically extract features and estimate conditions of the tested bearing. The domain adaptation module is used to learn a domain invariant space that minimizes the distribution discrepancy between different domains. From the results, it can be seen that the performance of the model can be improved as the labeled samples increase in training. The proposed method achieves better accuracies than the softmax-based method, which demonstrates the robustness of the learned prototypes. Generally, unsupervised DA-based methods need complex networks to perform better than the proposed method. Due to the lack of prior knowledge in the target domain, these methods need massive unlabeled target samples for training.

To visualize the effects of DA on the distribution of features from the source and target domains, we use *t-SNE* to map the high dimensional features into a two-dimensional space. The results of the transfer experiment $\mathcal{D}_U \rightarrow \mathcal{D}_S$ are shown in Fig. 6. From the results, it can be seen that the proposed method can reduce the inter-domain discrepancy of distributions. Compared with other methods, the learned features of the proposed method are inter-class separable, which makes the features more discriminative. It validates that the proposed method can reduce the distribution discrepancies of data obtained from different machines.

Conclusion

In this paper, we have proposed a CNN-based method in combination with DA and prototype learning for fault diagnosis. The proposed method takes raw vibration signals as inputs and achieves high classification accuracies on four transfer tasks. The experiments on two popular bearing datasets show effective transfer performance when a few labeled target samples are available. Compared with existing DA-based methods, the proposed method can be applied to address the problem when the classes from the source and target domains are partially overlapping. With only a few labeled target samples in training, the model can distinguish target classes and reject the samples from unknown source classes. Moreover, the model quickly converges with high accuracy as the labeled target samples increase in training. In future work, we will utilize other metrics for similarity measurement instead of Euclidean distance and increase the number of prototypes for each class. Overall, the effectiveness of

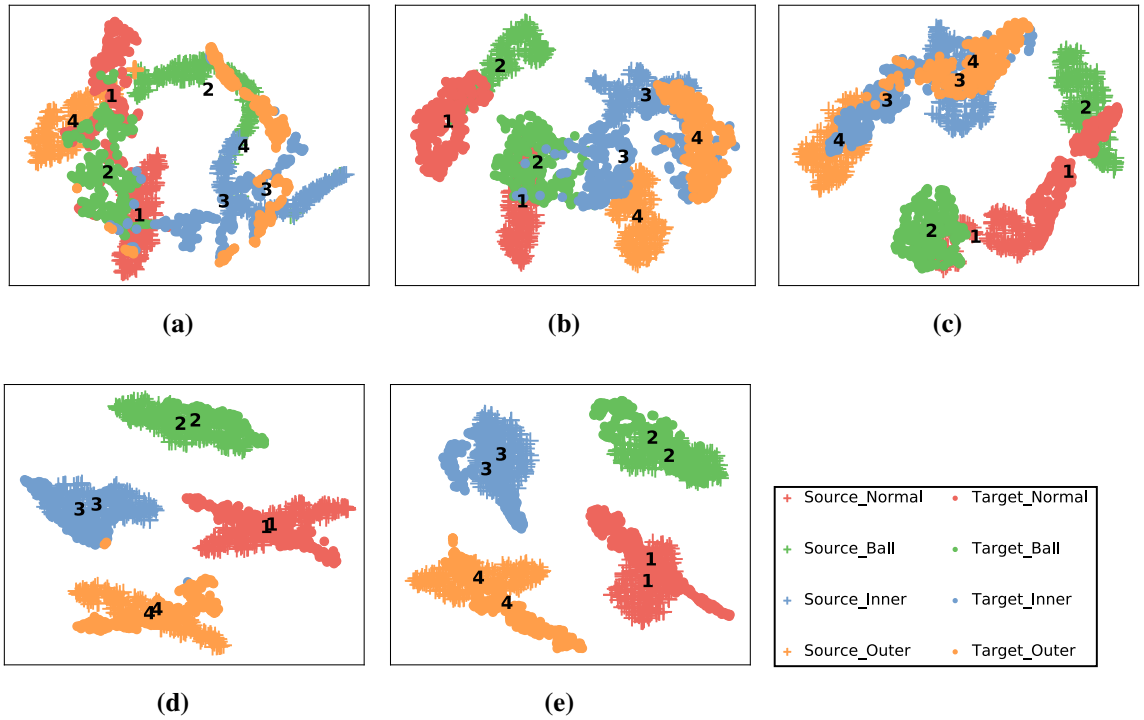

Fig. 6 *t-SNE* visualization of features: **a** Handcrafted features. **b** TCA-CNN. **c** DDC. **d** DSDAN. **e** DPDAN

the proposed method makes it a promising method for fault diagnosis.

Acknowledgements This work is supported by the National Key Research and Development Program (CN) under Grant 2018YFB1703 400 and the National Natural Science Foundation of China under Grants U1801263 and U1701262.

References

Ai, X., (2013). Rolling element bearings, history. In *Encyclopedia of tribology*, pp. 2932–2937. https://doi.org/10.1007/978-0-387-92897-5_331.

Bezdek, J. C., Reichherzer, T. R., Lim, G. S., & Attikiouzel, Y. (1998). Multiple-prototype classifier design. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *28*(1), 67–79. https://doi.org/10.1109/5326.661091.

Chen, X., Zhang, B., & Gao, D. (2020). Bearing fault diagnosis base on multi-scale cnn and lstm model. *Journal of Intelligent Manufacturing*,. https://doi.org/10.1007/s10845-020-01600-2.

Gao, Y., Gao, L., Li, X., & Zheng, Y. (2019). A zero-shot learning method for fault diagnosis under unknown working loads. *Journal of Intelligent Manufacturing*,. https://doi.org/10.1007/s10845-019-01485-w.

Gretton, A., Borgwardt, K., Rasch, M., Schölkopf, B., & Smola, A. J. (2007). A kernel method for the two-sample-problem. In: *Advances in neural information processing systems*, pp. 513–520.

Guo, L., Lei, Y., Xing, S., Yan, T., & Li, N. (2018). Deep convolutional transfer learning network: A new method for intelligent fault diagnosis of machines with unlabeled data. *IEEE Transactions on Industrial Electronics*, *66*(9), 7316–7325. https://doi.org/10.1109/TIE.2018.2877090.

Han, T., Liu, C., Yang, W., & Jiang, D. (2020). Deep transfer network with joint distribution adaptation: A new intelligent fault diagnosis framework for industry application. *ISA Transactions*, *97*, 269–281. https://doi.org/10.1016/j.isatra.2019.08.012.

Hassen, M., & Chan, P. K. (2020). Learning a neural-network-based representation for open set recognition. In *Proceedings of the 2020 SIAM international conference on data mining, SIAM*, pp. 154–162. https://doi.org/10.1137/1.9781611976236.18.

He, K., Zhang, X., Ren, S., & Sun, J.,(2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034. https://doi.org/10.1109/ICCV.2015.123.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R., (2012). Improving neural networks by preventing co-adaptation of feature detectors. arXiv:1207.0580.

Hoffman, J., Rodner, E., Donahue, J., Darrell, T., & Saenko, K., (2013). Efficient learning of domain-invariant image representations. arXiv:1301.3224.

Hsu, C. W., & Lin, C. J. (2002). A comparison of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks*, *13*(2), 415–425. https://doi.org/10.1109/72.991427.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv:1412.6980.

Kohonen, T. (1990). The self-organizing map. *Proceedings of the IEEE*, *78*(9), 1464–1480. https://doi.org/10.1007/978-3-662-62027-4_5.

Li, X., Zhang, W., & Ding, Q. (2018a). Cross-domain fault diagnosis of rolling element bearings using deep generative neural networks. *IEEE Transactions on Industrial Electronics*, *66*(7), 5525–5534. https://doi.org/10.1109/TIE.2018.2868023.

Li, X., Zhang, W., & Ding, Q. (2018b). A robust intelligent fault diagnosis method for rolling element bearings based on deep distance metric learning. *Neurocomputing*, *310*, 77–95. https://doi.org/10.1016/j.neucom.2018.05.021.

Li, X., Zhang, W., Ding, Q., & Sun, J. Q. (2019). Multi-layer domain adaptation method for rolling bearing fault diagnosis. *Signal Processing*, *157*, 180–197. https://doi.org/10.1016/j.sigpro.2018.12.005.

Liu, C. L., & Nakagawa, M. (2001). Evaluation of prototype learning algorithms for nearest-neighbor classifier in application to handwritten character recognition. *Pattern Recognition*, *34*(3), 601–615. https://doi.org/10.1016/S0031-3203(00)00018-2.

Liu, H., Zhou, J., Xu, Y., Zheng, Y., Peng, X., & Jiang, W. (2018). Unsupervised fault diagnosis of rolling bearings using a deep neural network based on generative adversarial networks. *Neurocomputing*, *315*, 412–424. https://doi.org/10.1016/j.neucom.2018.07.034.

Lu, W., Liang, B., Cheng, Y., Meng, D., Yang, J., & Zhang, T. (2016). Deep model based domain adaptation for fault diagnosis. *IEEE Transactions on Industrial Electronics*, *64*(3), 2296–2305. https://doi.org/10.1109/TIE.2016.2627020.

Maaten, Lvd, & Hinton, G. (2008). Visualizing data using t-sne. *Journal of Machine Learning Research*, *9*, 2579–2605.

Motiian, S., Piccirilli, M., Adjeroh, D. A., & Doretto, G. (2017). Unified deep supervised domain adaptation and generalization. In *Proceedings of the IEEE international conference on computer vision*, pp. 5715–5725. https://doi.org/10.1109/ICCV.2017.609.

Pan, S. J., Tsang, I. W., Kwok, J. T., & Yang, Q. (2010). Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, *22*(2), 199–210. https://doi.org/10.1109/TNN.2010.2091281.

Qiu, H., Lee, J., Lin, J., & Yu, G. (2006). Wavelet filter-based weak signature detection method and its application on rolling element bearing prognostics. *Journal of Sound and Vibration*, *289*(4–5), 1066–1090. https://doi.org/10.1016/j.jsv.2005.03.007.

Saito, K., Kim, D., Sclaroff, S., Darrell, T., & Saenko, K. (2019). Semi-supervised domain adaptation via minimax entropy. In *Proceedings of the IEEE international conference on computer vision*, pp. 8050–8058. https://doi.org/10.1109/ICCV.2019.00814.

Sato, A., & Yamada, K. (1996). Generalized learning vector quantization. In *Advances in neural information processing systems*, pp. 423–429.

Sato, A., & Yamada, K. (1998). A formulation of learning vector quantization using a new misclassification measure. In *Proceedings. Fourteenth international conference on pattern recognition (Cat. No. 98EX170)*, IEEE, vol. 1, pp. 322–325. https://doi.org/10.1109/ICPR.1998.711145.

Shimodaira, H. (2000). Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, *90*(2), 227–244. https://doi.org/10.1016/S0378-3758(00)00115-4.

Smith, W. A., & Randall, R. B. (2015). Rolling element bearing diagnostics using the case western reserve university data: A benchmark study. *Mechanical Systems and Signal Processing*, *64*, 100–131. https://doi.org/10.1016/j.ymssp.2015.04.021.

Snell, J., Swersky, K., & Zemel, R. (2017). Prototypical networks for few-shot learning. In *Advances in neural information processing systems*, pp. 4077–4087.

Sreejith, B., Verma, A., & Srividya, A. (2008). Fault diagnosis of rolling element bearing using time-domain features and neural networks. In *2008 IEEE region 10 and the third international conference on industrial and information systems*, IEEE, pp. 1–6. https://doi.org/10.1109/ICIINFS.2008.4798444.

Tong, Z., Li, W., Zhang, B., Jiang, F., & Zhou, G. (2018). Bearing fault diagnosis under variable working conditions based on domain adaptation using feature transfer learning. *IEEE Access*, *6*, 76187–76197. https://doi.org/10.1109/ACCESS.2018.2883078.

Tzeng, E., Hoffman, J., Zhang, N., Saenko, K., & Darrell, T. (2014). Deep domain confusion: Maximizing for domain invariance. arXiv:1412.3474.

Wang, M., & Deng, W. (2018). Deep visual domain adaptation: A survey. *Neurocomputing*, *312*, 135–153. https://doi.org/10.1016/j.neucom.2018.05.083.

Wen, L., Gao, L., & Li, X. (2017a). A new deep transfer learning based on sparse auto-encoder for fault diagnosis. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *49*(1), 136–144. https://doi.org/10.1109/TSMC.2017.2754287.

Wen, L., Li, X., Gao, L., & Zhang, Y. (2017b). A new convolutional neural network-based data-driven fault diagnosis method. *IEEE Transactions on Industrial Electronics*, *65*(7), 5990–5998. https://doi.org/10.1109/TIE.2017.2774777.

Yang, H. M., Zhang, X. Y., Yin, F., & Liu, C. L. (2018). Robust classification with convolutional prototype learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3474–3482. https://doi.org/10.1109/CVPR.2018.00366.

Yu, Y., Junsheng, C., et al. (2006). A roller bearing fault diagnosis method based on emd energy entropy and ann. *Journal of Sound and Vibration*, *294*(1–2), 269–277. https://doi.org/10.1016/j.jsv.2005.11.002.

Zhang, A., Li, S., Cui, Y., Yang, W., Dong, R., & Hu, J. (2019). Limited data rolling bearing fault diagnosis with few-shot learning. *IEEE Access*, *7*, 110895–110904. https://doi.org/10.1109/ACCESS.2019.2934233.

Zhang, W., Peng, G., Li, C., Chen, Y., & Zhang, Z. (2017). A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals. *Sensors*, *17*(2), 425. https://doi.org/10.3390/s17020425.

Zhao, K., Jiang, H., Wu, Z., & Lu, T. (2020). A novel transfer learning fault diagnosis method based on manifold embedded distribution alignment with a little labeled data. *Journal of Intelligent Manufacturing*,. https://doi.org/10.1007/s10845-020-01657-z.