Hindawi
Complexity
Volume 2022, Article ID 4598725, 15 pages
https://doi.org/10.1155/2022/4598725

WILEY | Hindawi

Research Article

A Diagnosis Framework for High-reliability Equipment with Small Sample Based on Transfer Learning

Jinxin Pan, Bo Jing, Xiaoxuan Jiao, Shenglong Wang, and Qingyi Zhang

[1]*Air Force Engineering University, Xi'an/710038, China*

Correspondence should be addressed to Jinxin Pan; panjinxin_sensor@126.com

Received 17 August 2021; Revised 14 September 2021; Accepted 2 December 2021; Published 23 February 2022

Academic Editor: Kiyong Oh

Conventional methods for fault diagnosis typically require a substantial amount of training data. However, for equipment with high reliability, it is arduous to form a large-scale well-annotated dataset due to the expense of data acquisition and costly annotation. Besides, the generated data have a large number of redundant features which degraded the performance of models. To overcome this, we proposed a feature transfer scenario that transfers knowledge from similar fields to enhance the accuracy of fault diagnosis with small sample. To reduces the redundant information, data were filtered according to manifold consistency. Then, features were extracted based on CNN and feature transfer was conducted. For adequate fitness, the joint adaptation of conditional distribution and marginal distribution was used between the two domains. Minimum structural risk and MMD of adaptation were two indicators weighted for training the model. To test the efficiency of the model, we built an airborne fuel pump testbed, and contributed a new dataset that contained 15 categories of fault data, which serves as the small sample dataset in this research. Then the proposed model was applied in our experimental data. As a result, the fault diagnosis rate increases by 28.6% through our proposed model, which is more precise than other classical methods. The results of feature visualization further demonstrate that the features are more distinguished through the proposed method. All code and data are accessible on my GitHub.

1. Introduction

In recent years, data-driven methods have been widely applied in the fields of prognostics and health management, which have become a hot method in building complex diagnosis models [1–3]. Most data-driven methods are based on a large number of samples, from which the corresponding relationship between input and output is extracted to establish a model. However, for some equipment with high reliability and long lifespan, it is arduous to obtain sufficient fault data. Along with that, the generated data have a large number of redundant features which degraded the performance of models. A model trained by small samples has limited generalization ability, which will lead to low accuracy when applied to other fields [4].

Some popular methods have been proposed to solve the problem of small sample. To our best knowledge, these methods are of three categories. The first is based on resample, which resampling small samples to generate more data, such as Random Under-sampling [5] and Synthetic Minority Over-sampling [6]. The second is based on Generative Adversarial Net [7, 8], whose principle is to make the generated network sample as realistic as possible by antagonizing generative network and discriminant network, so as to enlarge small samples. The third is based on few-shot learning [9, 10], which decompose the small samples into different meta tasks to learn the generalization ability of the model. Therefore, few-shot learning has adaptive capacity to an unseen dataset. Although these methods achieved success to some extent, their sources of knowledge are only from the small samples. From the perspective of information theory, these methods do not change the nature of a small sample with little knowledge.

Transfer learning is a new machine learning method that uses existing knowledge to solve problems in different but related fields [11]. Transfer learning has been applied to some tasks such as hand gesture recognition [12], sentiment analysis [13], fraud detection [14], and hyperspectral image analysis [15]. Besides, many advanced transfer learning theories have been proposed. For example, Liu [16]

proposed a one-step approach towards classifiers have to be trained with noisy data. Chen [17] proposed a boundary based Out-of-Distribution (OOD) classifier which classifies the unseen and seen domains by only using seen samples for training. Teshima [18] proposed a meta-distributional scenario in which a data generating mechanism is invariant among domains.

Due to the advantages of transfer learning in domain generalization, it is also widely applied in fault diagnosis. Wang [19] proposed an LDA-based deep transfer learning framework for fault diagnosis in industrial chemical processes, Singh [20] utilized minimum redundancy maximum relevance (mRMR) for intelligent fault diagnosis of rotating machines. Deng [21] proposed a double-layer attention based adversarial network (DA-GAN) for partial transfer learning in machinery fault diagnosis. To draw a conclusion from the existing literatures, there are two aspects that few researchers had considered. Firstly, most proposed transfer learning methods focus on the adaptation of the diagnosis algorithm in different fields but pay less attention to the condition of small samples. Secondly, most literatures use the raw signal in the source domain, which is feasible for many fields. However, some high-reliability equipment has a long lifespan, which results in redundant features of monitoring data. If the raw signal is used directly, negative migration may occur.

Given that high-reliability equipment has characteristics of small sample size and redundant features, we proposed a feature transfer framework. To reduces the redundant information, data were filtered according to manifold consistency. Then, features were extracted based on CNN and feature transfer was conducted. For adequate fitness, the joint adaptation of conditional distribution and marginal distribution was used between the two domains. Minimum structural risk and MMD of adaptation were two indicators weighted for training the model to enhance the generalization ability of the model. To test the efficiency of the model, we built an airborne fuel pump testbed, and contributed a new dataset that contained 15 categories of fault data, which serves as the small sample dataset in this research. Then the proposed model was applied in our experimental data. As a result, the fault diagnosis rate increases by 28.6% through our proposed model, which is more precise than other classical methods. The results of feature visualization further demonstrate that the features are more distinguished through the proposed method.

2. Problem Setup

2.1. Definition of transfer learning. In most cases, a domain D consists of two components: a feature space χ and a marginal probability distribution $P(X)$, where $X = \{x_1, ..., x_n\} \in \chi$. For example, given a concrete domain $D = \{X, P(X)\}$, a task can be expressed as two components: a label space Υ and an objective predictive function $f(\cdot)$ (denoted by$T = \{Y, f(\cdot)\}$), which is not observed but can be trained from the data, which consist of pairs $\{x_i, y_i\}$, where $x_i \in \chi$ and$y_i \in \Upsilon$. The function $f(\cdot)$ can be used to predict from the data to the corresponding label$f(x)$. From a probabilistic viewpoint, $f(x)$ can be written as$P(y \mid x)$.

Definition of transfer learning: Given a source domain D^s and learning taskT^s, a target domain D^t and learning taskT^t, transfer learning aims to help improve the learning of the target predictive function $f^t(\cdot)$ inD^t using the knowledge in D^s andT^s, where $D^s \neq D^t$, or $T^s \neq T^t$[22].

2.2. Fault diagnosis with small sample. For some highly reliable products, the small sample data set can be represented as $\{x_i^t, y_i^t\}_{i=1}^{n_t}$, which contains n_t samples. In the form of transfer learning, the data set is described as target domain $D^t = \{x^t, P(x^t)\}$ and the target task $T^t = \{y^t, f^t(\cdot)\}$, where $P(x^t)$ is the Marginal Distribution of x^t, y^t is the label space of the target domain, $f^t(\cdot)$ is a function that maps the sample x^t to the tag space y^t in the target domain.

Another dataset with rich samples is represented as $\{x_i^s, y_i^s\}_{i=1}^{n_s}$, which contains n_s samples ($n_s \gg n_t$). In the form of transfer learning, the data set is described as the source domain $D^s = \{x^s, P(x^s)\}$ and the source task $T^s = \{y^s, f^s(\cdot)\}$ in transfer learning, where $P(x^s)$ is the marginal distribution of x^s, y^s is the label space of the source domain, $f^s(.)$ is a function that maps the sample x^s to the tag space y^s in the source domain.

The goal of transfer learning is to acquire and apply the knowledge from the source domain [23]. More specifically, it is to establish the nonlinear mapping relationship from the equipment monitoring data to the health label space in the source domain, then transfer it to the target domain. In the given situation, the labels of the source domain and target domain are accessible. Such problem is concluded as multitask learning. Aiming at this problem, feature transfer is often used to transfer knowledge from source domain to target domain [22].

2.3. Feature transfer. The idea of feature transfer is to learn a pair of mapping functions $\{\varphi^s(\cdot), \varphi^t(\cdot)\}$ to extract diagnostic features respectively from the source domain and the target domain. Then adapt features extracted by mapping functions, and the target domain could extract features according to the paradigm of the source domain. The fault classification is carried out in the target domain based on the features, and the diagnosis knowledge of the source domain is transferred to the target domain through the feature adaptation [24].

Aiming for learning transferable features between a given target domain and source domain, a common approach of feature adaptation is to minimize inter-domain differences between source feature and target feature. During the adaptation, the extracted features from the target domain act as a template, that the source domain could learn from the template. The schematic diagram of domain adaptation is shown in Figure 1.

3. General Framework

The main issues of transfer learning can be concluded as the following three: when to transfer, what to transfer and how to transfer. Aiming at these three issues, this paper designed a framework of LLE-CNN-JDA. Aiming at when to transfer, we designed a data filtering method based on manifold

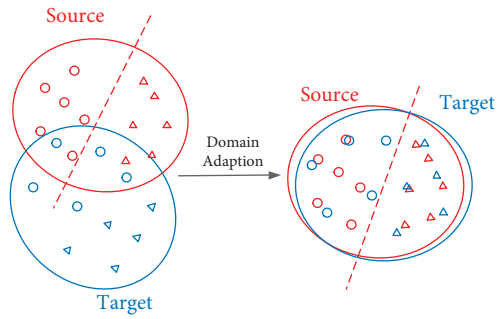


Figure 1: Schematic diagram of domain adaptation.

consistency. We mapped the high-dimensional data into low dimensional space, to analyze the similarity between the source domain and target domain. To ensure the availability of transfer data, we filtered data in the source domain based on the Euclidean distance between the source domain and the target domain in manifold space. Aiming at what to transfer, the method of feature transfer is adopted, the convolutional neural network is used to extract the deep feature of the data in the source domain and target domain. By adapting the feature in the source domain and target domain, the knowledge may be transferred in the feature layer. Aiming at how to transfer, we designed a term that jointly adapt conditional distribution and marginal distribution of feature layer, and the network is trained based on structural risk minimization. The framework diagram of the proposed model is shown in Figure 2.

3.1. Data filtered based on manifold consistency. The so-called manifold is a general term for geometric objects, such as curves and surfaces of various dimensions. Manifold learning maps data from higher-dimensional space into lower-dimensional space. Unlike other dimensionality reduction methods, manifold learning assumes that data is sampled from a potential manifold. If we can find the laws of the data in the manifold space, we may find the potential laws of the data in high dimensions to mine the essential characteristics of data [25, 26].

For the given situation, the sample size of the source domain is large, so the prognostics model in the source domain can be trained well. However, the data in the target domain is not abundant, we may not excavate enough information from the target domain. The source domain has sufficient data, but it is necessary to explore whether the source data can be effectively applied to the target domain. Based on the idea of manifold learning, we consider that if we can explore the relationship between the source domain and the target domain in the low-dimensional manifold space, and filter the data transferable to the target domain, source data may be applicable to the target domain.

Locally linear embedding (LLE) is a method of manifold learning, which enables the data to maintain the original manifold structure well after dimension-reduction. The manifold of LLE is an unclosed surface, which has features of relatively uniform and dense distribution. Every data point can be constructed by the linear weighted combination of its nearest points. LLE transfers manifolds from higher dimensions to lower dimensions and preserves some features of manifolds in higher dimensions as much as possible [27, 28]. The steps of the LLE algorithm are as follows:

Step1. :Calculate K adjacent points for each sample point. Adopting the KNN strategy, K points with the smallest Euclidean distance to the sample point are taken as the K adjacent points of the sample.

Step2. :Calculate the local reconstruction weight matrix W of the sample. The reconstruction error is defined as $\varepsilon(W) = \sum_i |\overrightarrow{X_i} - \sum_j W_{ij}\overrightarrow{X_j}|^2$, and the local covariance matrix C is defined as $C_{jk} = (\overrightarrow{x} - \overrightarrow{\eta}_j) \cdot (\overrightarrow{x} - \overrightarrow{\eta}_k)$, where X represents a specific point, and its K adjacent points are denoted by η. Then Minimize $\varepsilon(W)$, then get $W_j = \sum_k C_{jk}^{-1} / \sum_{lm} C_{lm}^{-1}$.

Step3. :Map all sample points to a low-dimensional space, where the mapping function satisfies $\min_Y \Phi(Y) = \sum_i \overrightarrow{Y_i} \sum_j | - \sum_j W_{ij}\overrightarrow{Y_j}|^2$. The formula can also be represented as $\Phi(Y) = \sum_{ij} M_{ij}(\overrightarrow{Y_i} \cdot \overrightarrow{Y_j})$, where $M = (I - W)^T(I - W)$. Combining with the restrictive conditions $\sum_i Y = \overrightarrow{0}$ and $1/N\sum_i Y_i \overrightarrow{Y_i}^T = I$, the problem is transformed into MY = λY, take M eigenvectors of matrix M to form column vectors, that matrix Y = N ∗ M, where N is the size of data.

In this paper, we adopted the LLE algorithm to evaluate the similarity of the data from the source domain and target domain. We adopted the bearing failure data from Case Western Reserve University as the source domain, the failure data from our airborne fuel pump test-bed as the target domain. Two kinds of data were mapping to manifold space respectively. As the neighbouring points number K change, the mapping results in manifold space are shown in Figure 3. Through analysis, we found that when K = 100, the mapping manifold of the two types of data is most close. Therefore, the K = 100 was chosen to filter data in the low dimension. The failure data from the airborne fuel pump testbed worked as the target domain, which was in a small sample size. The bearing failure data from Case Western Reserve University worked as the source domain. The bearing failure data is in big sample size, but it is difficult to ensure the validity of the data. Therefore, we proposed to filter data of the source domain in the manifold space by calculating the Euclidean distance to the target domain. We adopted the tactics of KNN, computed the distance of each sample point from the source domain to all sample points from the target domain, then chose the minimum distance $d_i = \min|X_i^s - X_j^t|\,(j = 1 \cdots n_t)$ as an indicator of the sample point. We got an indicator set $\{d_i\}(i = 1 \cdots n_s)$, then sorted the indicator set, and chose the **n** sample points with the smallest distance as the target domain.

3.2. Deep feature extraction. The source domain and target domain data both contain prognostics information related to the equipment. Therefore, we adopt a convolutional

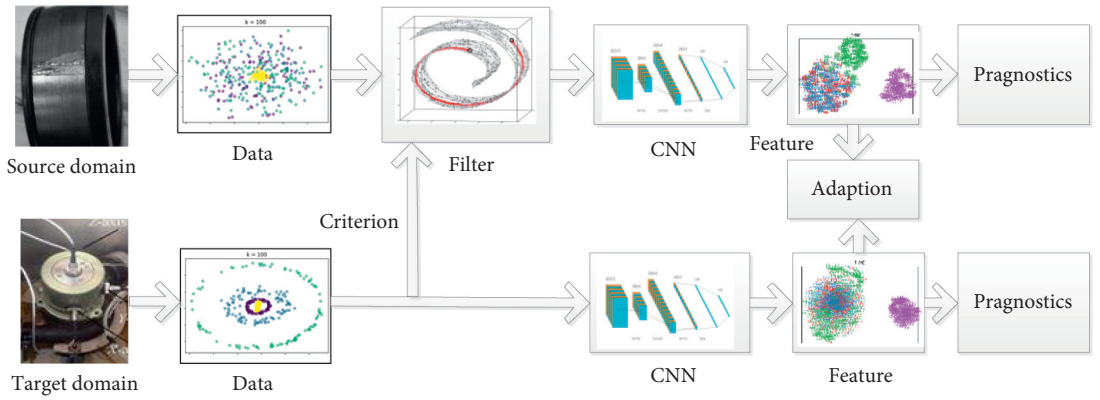


Figure 2: The framework of the proposed model.

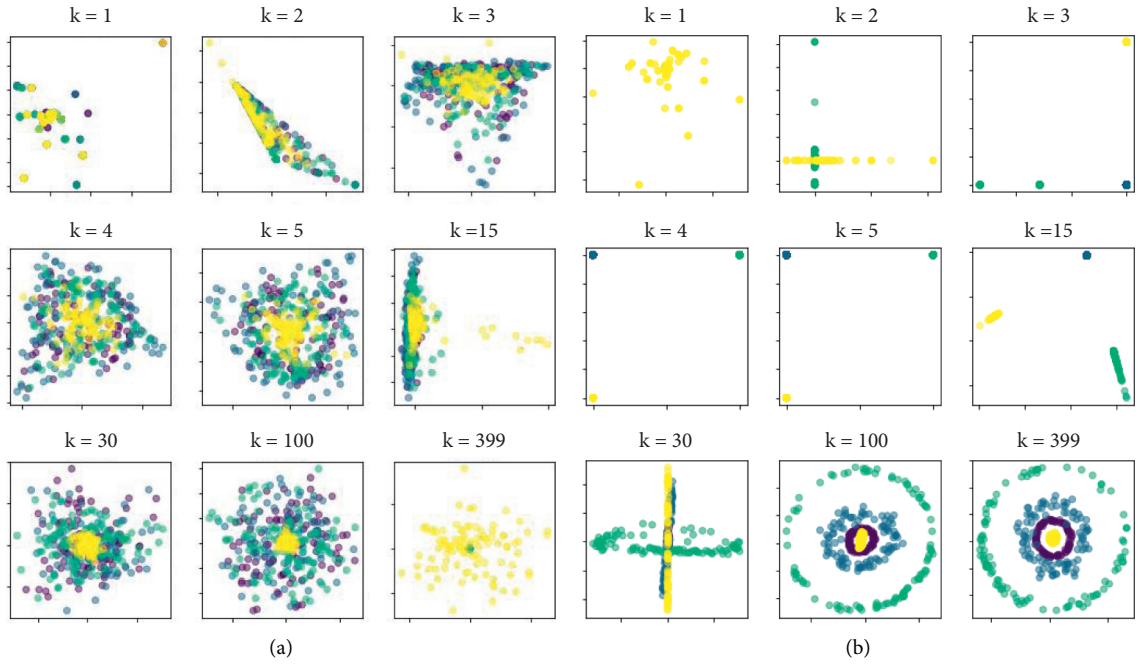


Figure 3: Results of LLE under the different value of K; (a) LLE of bearing fault data; (b) LLE of airborne fuel pump fault data.

neural network to extract fault features. The source domain data and target domain data were extracted separately by the neural network, which contains convolution layer, pooling layer, flatten layer and full connection layer. The network parameters are trained based on structural risk minimization and feature adaptation between the source domain and target domain.

Convolution layer operation: Assume that $x_i^{s,l-1}$ and $x_i^{t,l-1}$ represent vectors in the L-1 layer of the network. Through a convolution kernel function $k \in R^{16\times3}$, the feature vector of layer L is $x_i^{s,l} = F(x_i^{s,l-1}; \theta^l) = \sigma_l(x_i^{s,l-1} * k^l + b^l)$, where σ_l is the activation function of layer L, $\theta^l = \{k^l, b^l\}$ is the parameters to be trained of the convolution layer.

Pooling layer operation: The processing logic of the pooling layer is to compress the input matrix. The formula of the pooling layer is $v_{m,n}^{s,l} = \max\{x_i^{s,l} | m \le i \le n\}$, where $v_{m,n}^{s,l}$ represents the result of pooling the features from sequence M to N in the convolution layer L, $x_i^{s,l}$ represents the vector whose sequence is i in the convolution layer L, s represents that the operation is for the source domain.

After multiple convolution and pooling processes, the feature extraction layer output, namely the flatten layer input vector $x_i^{s,C}$, is obtained. The vector fed into flatten function, and the flatten layer output $x_i^{s,F} = \text{flatten}(x_i^{s,C})$ was obtained. The output of the flattening layer is then used as the input vector of the full connection layer, where the vector is mapped to the label space through the neural network of the full connection layer. The function is $y_i^{s,o} = g(x_i^{s,F}; \theta^F) = \sigma(w^F \cdot x_i^{s,F} + b^F)$, where σ is the activation function, $\theta^F = \{w^F, b^F\}$ is the parameter set to be trained.

3.3. Structural risk minimization. For the target domain data with a small size, overfitting is easy to occur in the training process. To prevent overfitting, it is necessary to avoid the excessive complexity of network structure in the process of

training. Therefore, complexity and accuracy are important indicators impacting the efficiency of the network. In this paper, the Convolutional Neural Network was trained based on structural risk minimization. A penalty term (regularization term) for the complexity of the model is added to the empirical risk to reduce the risk of data overfitting [29]. The formula of structural risk minimization is expressed as follows:

$$\min_{f\in F} \frac{1}{N}\sum_{i=1}^{N} L\left(y_i, f\left(x_i\right)\right) + \lambda J\left(f\right). \tag{1}$$

In the formula, the more complex the model f is, the greater the value of $J(f)$ will be. λ indicates the importance of model complexity. As the empirical risk convergence to a certain degree, when the empirical risk decreases, model complexity will increase sharply. Model complexity may make the model fit the data in the source domain over exactly, and the model would be difficult to generalize to the target domain. Thus, we added the penalty term for model complexity to inhibits the excessive increase of model complexity.

For a conditional probability distribution, the loss function is logarithmic, and the model complexity is determined by the prior probability of the model. Therefore, the structural risk minimization is equal to the maximum posterior probability estimate. Given a sample set $T = \{(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)\}$, it is assumed that the prior distribution of parameter θ is$g(\theta)$, and the probability of T is$\prod_{i=1}^{N} P(y_i \mid x_i; \theta)g(\theta)$. Maximize the probability as $\max_{\theta} \prod_{i=1}^{N} P(y_i \mid x_i; \theta)g(\theta)$, then take the log of the result, $\max_{\theta}\left\{\sum_{i=1}^{N} \log P(y_i \mid x_i; \theta) + \log g(\theta)\right\}$ will be obtained. Take the complex number of the above equation, it is transformed into the minimization problem $\min_{\theta}\left\{\sum_{i=1}^{N} -\log P(y_i \mid x_i; \theta) + \log 1/g(\theta)\right\}$. Define the loss function as $L(x_i, P(y_i \mid x_i; \theta)) = -\log P(y_i \mid x_i; \theta)$, the coefficient as $\lambda = 1/N$, the penalty term as $J(f) = \log 1/g(\theta)$, the equation is equal to form (2), which is structural risk minimization [30].

$$\min_{\theta}\left\{\frac{1}{N}\sum_{i=1}^{N} L\left(y_i, P\left(y_i \mid x_i; \theta\right)\right) + \lambda J\left(f\right)\right\}. \tag{2}$$

3.4. Joint adaption of conditional distribution and marginal distribution. Feature extraction and network training are conducted separately in the source domain and target domain. To transfer knowledge, the adaptation between the source domain and the target domain is to be conducted at the feature layer.

As for the source domain, the size of data is large, so the extracted feature contains a lot of information related to the equipment. Thus, the pattern how deep features be extracted in source domain is an empirical model for fault identifying, which may be appliable to other related fields. For the target domain, the sample size is small, which may lead to poor generalization ability of the network in feature extraction. Therefore, the feature extraction of the source domain is a reference to the target domain, and feature transfer will be available in this way.

Marginal distribution and conditional distribution reflect domain distribution [31]. Therefore, feature adaptation is to adapt marginal distribution and conditional distribution. According to probability theory $J = P \cdot Q$, we seek to minimize the distribution distance (1). between the marginal distributions P^s andP^t, and (2). between the conditional distributions Q^s andQ^t simultaneously. For source domain $D^s = \{X^s, P(X^s)\}$ and target domain $D^t = \{X^t, P(X^t)\}$, assume that the features extracted through CNN network are $x_i^{s,C}$ and $x_i^{t,C}$. If the features of the two domains are to be adapted, the marginal distribution and conditional distribution of feature vectors must be adapted.

3.4.1. Adaption of the marginal distribution. We try to minimize the distance between marginal distributions P^s andP^t. Since directly estimating probability densities is nontrivial, we resort to explore nonparametric statistics. We adopt empirical Maximum Mean Discrepancy (MMD) to measure the distance, which compares different distributions based on the distance between the sample means of two domains in a reproducing kernel Hilbert space (RKHS) [32].

Specifically, the statistical approach of MMD is conducted in the following manner. Based on the samples of the two distributions, look for a continuous function $f(\cdot)$ in the sample space, get the function values corresponding to the two distributions, and calculate the mean of the function values of each distribution. By making difference between the two mean values, the mean discrepancy of the two distributions will be obtained corresponding to$f(\cdot)$. Look for an $f(\cdot)$ that causes the mean discrepancy to have a maximum value, the value is MMD. Thus, MMD is taken as a test statistic to determine whether the two distributions were close. If this value is small enough, the two distributions are considered the same; otherwise, they are not. This value is also used to determine the degree of similarity between two distributions [33]. If F is used to represent a continuous set of functions in the sample space, then MMD can be expressed as follows:

$$\mathrm{MMD}[F, p, q] = \sup_{f\in F}\left(E_{x\sim p}[f(x)] - E_{y\sim p}[f(y)]\right). \tag{3}$$

Assume that X and Y are two data sets obtained by independent identical distribution sampling from distribution p and q respectively, and the sizes of the data sets are M and N. The empirical estimate of MMD based on X and Y is as follows:

$$\mathrm{MMD}[F, p, q] = \sup_{f\in F}\left(\frac{1}{m}\sum_{i=1}^{m} f\left(x_i\right) - \frac{1}{n}\sum_{i=1}^{n} f\left(y_i\right)\right). \tag{4}$$

Given two distributions of observation sets X, Y, this result will depend heavily on a given set of functions F. To express the properties of MMD, if and only if P and Q are of the same distribution, MMD is 0. Thus, F is required to be rich enough. On the other hand, the empirical estimation of MMD should rapidly converge to its expectation as the size

of the observation set increases. Thus, F must be sufficiently restrictive. To satisfy the two requirements, we adopt the reproducing kernel Hilbert Spaces.

In reproducing kernel Hilbert Spaces, F space is a complete inner product space, and each F corresponds to a feature map. Based on the feature map, we defined a mean embedding of p for a distribution p that satisfies the following properties: $\mu_p \in H$ such that $E_x(f) = \langle f, \mu_p \rangle_H$ for all $f \in H$. Mean embedding exists with constraints. Under the existence of the mean embedding of P and Q, the MMD squared can be expressed as follows:

$$\begin{aligned} \mathrm{MMD}^2[F, p, q] &= \left[\sup_{\|f\|_H \le 1} \left(E_x[f(x)] - E_y[f(y)] \right) \right]^2 \\ &= \left\| \mu_p - \mu_q \right\|_H^2. \end{aligned} \tag{5}$$

If F is a unit ball in a universal RKHS, such as Gaussian and Laplace RKHSs, the square of this MMD can be expressed as:

$$\begin{aligned} \mathrm{MMD}^2[F, p, q] = E_{x,x'}[k(x, x')] - 2E_{x,y}[k(x, y)] \\ + E_{y,y'}[k(y, y')]. \end{aligned} \tag{6}$$

X and X 'are two random variables that obey p, and y and y' are two random variables that obey q. One of the above statistical estimates can be expressed as:

$$\mathrm{MMD}[F, X, Y] = \left[\frac{1}{m^2} \sum_{i,j=1}^{m} k(x_i, x_j) - \frac{2}{\mathrm{mn}} \sum_{i,j=1}^{m,n} k(x_i, y_j) + \frac{1}{n^2} \sum_{i,j=1}^{n} k(y_i, y_j) \right]^{1/2}. \tag{7}$$

Marginal distribution adaptation is to maximize mean discrepancies that for minimizing the marginal distributions of the features from the source domain and target domain [31], which is:

$$\min \mathrm{MMD}\left(x_i^{s,C}, x_i^{t,C}\right). \tag{8}$$

3.4.2. Adaption of the conditional distribution. We try to minimize the distance between conditional distributions Q^s andQ^t. Since calculating the nonparametric statistics of $Q^s(y^s|x^s)$ and$Q^t(y^t|x^t)$ are difficult, we resort to explore the quasi-conditional distributions as $Q^s(x^s|y^s)$ and $Q^t(x^t|y^t)$ instead, which can well approximate $Q^s(y^s|x^s)$ and$Q^t(y^t|x^t)$ when sample sizes are large [34, 35].

Conditional distribution adaptation is to maximize mean discrepancies. Simultaneously, the quasi-conditional distributions of the features from the source domain and target domain will reach a minimum, namely:

$$\min \mathrm{MMD}\left(x_i^{s,C}|y_k, x_i^{t,C}|y_k\right). \tag{9}$$

4. Training of the network

Overall, manifold consistency was used to filter data from the source domain. The deep features of the source domain and the target domain were extracted by CNN, then the features were adapted, and the classifier was constructed through the full connection layer. After network construction, network parameters need to be trained in the following aspects: (1). structural risk minimization; (2). marginal distribution adaptation; (3). conditional distribution adaptation. The network structure of the proposed model is shown in Figure 4. The joint optimization formula is as follows:

$$\begin{aligned} \min_{\theta} \frac{1}{N} \sum_{i=1}^{N} L\left(y_i^t, f\left(x_i^t\right)\right) + \lambda_t J\left(f^t\right) + \alpha \frac{1}{N} \sum_{i=1}^{N} L\left(y_i^s, f\left(x_i^s\right)\right) \\ + \lambda_s J\left(f^s\right), \\ + \beta \cdot \sum_{i=1}^{N} \mathrm{MMD}\left(x_i^{s,C}, x_i^{t,C}\right) + \delta \cdot \sum_{i=1,k=1}^{N,K} \mathrm{MMD} \\ \cdot \left(x_i^{s,C}|y_k, x_i^{t,C}|y_k\right). \end{aligned} \tag{10}$$

Due to the addition of the adaptive function in the feature layer, the parameter training and backpropagation of the entire convolutional neural network will be affected. For a single fault diagnosis with convolutional neural network, the error is from the difference between the expected label and the real label. However, when the joint distribution adaptation function is added to the feature layer, the adaptation error of the feature layer will also affect the parameter training of the whole network. Therefore, to search how the network is trained, the error backpropagation formula of the network is derived in this paper.

4.1. Error backpropagation in the full connection layer. The error of the output layer comes from the difference between the expected label and the real label, which is usually expressed by the two-norm of the difference between the two labels. The formula is as follows:

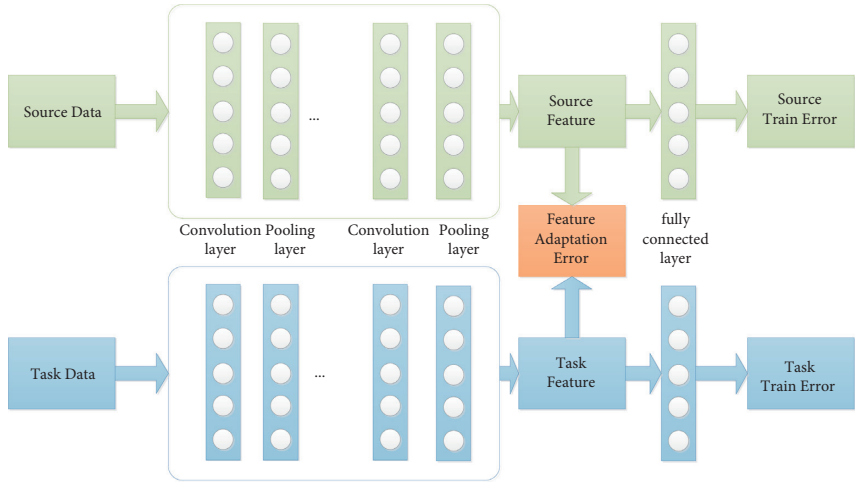


Figure 4: The network structure of the proposed model.

$$J(W,b,x,y)=\frac{1}{2}\left\|a^L-y\right\|_2^2, \tag{11}$$

Where a^L is output of the network, y is the label of the data set. According to the functional relationship, a^L can be expressed as: $a^L=\sigma(z^L)=\sigma(W^La^{L-1}+b^L)$, so we can get:

$$\begin{aligned}\frac{\partial J(W,b,x,y)}{\partial W^L}&=\left[\left(a^L-y\right)\odot\sigma'\left(z^L\right)\right]\left(a^{L-1}\right)^T,\\ \frac{\partial J(W,b,x,y)}{\partial b^L}&=\left[\left(a^L-y\right)\odot\sigma'\left(z^L\right)\right],\end{aligned} \tag{12}$$

Where $\odot$ is Hadamard product, σ is activation function, W^L and b^L are the weights and bias of the output layer.

In this paper, a single-layer full connection layer is used as a feature classifier. To enhance the extensibility of the algorithm, a more general case of multi-layer full connection is considered in the derivation of the backpropagation formula. Suppose an error propagation variable is:

$$\begin{aligned}\delta^l&=\frac{\partial J(W,b,x,y)}{\partial z^l}\\&=\left(\frac{\partial z^L}{\partial z^{L-1}}\frac{\partial z^{L-1}}{\partial z^{L-2}}\cdots\frac{\partial z^{l+1}}{\partial z^l}\right)^T\frac{\partial J(W,b,x,y)}{\partial z^L}.\end{aligned} \tag{13}$$

If we can figure out δ^l , according to the formula of layer l: $z^l=W^la^{l-1}+b^l$, the gradient formula of layer l can be obtained as follows:

$$\begin{aligned}\frac{\partial J(W,b,x,y)}{\partial W^l}&=\delta^l\left(a^{l-1}\right)^T,\\ \frac{\partial J(W,b,x,y)}{\partial b^l}&=\delta^l.\end{aligned} \tag{14}$$

So, the whole point of the problem is to figure out δ^l. In this paper, δ^l was deduced by mathematical induction. As for the output layer, $\delta^L=\partial J(W,b,x,y)/\partial z^L=(a^L-y)\odot\sigma'(z^L)$. For layer l, δ^l can be figured out throughδ^{l+1} from layer$l+1$, the formula is as follows:

$$\begin{aligned}\delta^l&=\left(\frac{\partial z^{l+1}}{\partial z^l}\right)^T\frac{\partial J(W,b,x,y)}{\partial z^{l+1}},\\&=\operatorname{diag}\left(\sigma'\left(z^l\right)\right)\left(W^{l+1}\right)^T\delta^{l+1},\\&=\left(W^{l+1}\right)^T\delta^{l+1}\odot\sigma'\left(z^l\right).\end{aligned} \tag{15}$$

Therefore, δ^l of each layer can be obtained by continuous backward recursion from the output layer, and then the updated formula of weight and bias of each layer can be calculated as:

$$\begin{aligned}W^l&=W^l-\alpha\sum_{i=1}^m\delta^{i,l}\left(a^{i,l-1}\right)^T,\\ b^l&=b^l-\alpha\sum_{i=1}^m\delta^{i,l}.\end{aligned} \tag{16}$$

4.2. Error backpropagation in pooling layer. There is no need to optimize and update W and B in the pooling process. However, in the process of backpropagation, the error will change in the pooling layer. Similar to the full connection layer, we still use δ^l as the bridge to calculate the backpropagation formula for the pooled layer.

For the pooling layer, during the backpropagation, we firstly restore all of the submatrix matrix sizes of δ^l to their pre-pooling sizes. If the pooling operation is Max, the values of each pooled locality of all submatrices of δ^l are placed at the position where the previous forward propagation algorithm obtained the maximum value. If the pooling operation is Average, then the values of each pooling locality of all the submatrices of δ^l are averaged and placed at the reduced submatrix position. This process is usually called the Upsample, and the formula is as follows:

$$\delta^l=\left(\frac{\partial a^l}{\partial z^l}\right)^T\frac{\partial J(W,b)}{\partial a^l}=\operatorname{upsample}\left(\delta^{l+1}\right)\odot\sigma'\left(z^l\right), \tag{17}$$

Where the Upsample function completes the logic of enlarging the pool error matrix and redistributing the error.

4.3. Error backpropagation in the convolution layer. Based on the analysis of the full connection layer, we can get that the recursion relation of δ^l in the convolution layer is:

$$\begin{aligned}\delta^l &= \frac{\partial J(W,b,x,y)}{\partial z^l} = \left(\frac{\partial z^{l+1}}{\partial z^l}\right)^T \frac{\partial J(W,b,x,y)}{\partial z^{l+1}} \\ &= \left(\frac{\partial z^{l+1}}{\partial z^l}\right)^T \delta^{l+1}.\end{aligned} \tag{18}$$

The key to the problem is to solve for $\partial z^{l+1}/\partial z^l$. According to the matrix relation: $z^{l+1} = a^l \cdot W^{l+1} + b^{l+1} = \sigma(z^l) \cdot W^{l+1} + b^{l+1}$, it can be obtained that: $\delta^l = (\partial z^{l+1}/\partial z^l)^T \delta^{l+1} = \delta^{l+1} \cdot \mathrm{rot180}(W^{l+1}) \odot \sigma'(z^l)$. According to the matrix relation: $z^l = a^{l-1} \cdot W^l + b^l$, the gradient of W and b can be obtained as follows:

$$\begin{aligned}\frac{\partial J(W,b,x,y)}{\partial W^l} &= a^{l-1} \cdot \delta^l, \\ \frac{\partial J(W,b,x,y)}{\partial b^l} &= \sum_{u,v} \left(\delta^l\right)_{u,v}.\end{aligned} \tag{19}$$

Thus, the updated formula of weight and bias of each layer can be calculated as follows:

$$\begin{aligned}W^l &= W^l - \alpha \sum_{i=1}^{m} a^{i,l-1} \cdot \delta^{i,l}, \\ b^l &= b^l - \alpha \sum_{i=1}^{m} \sum_{u,v} \left(\delta^{i,l}\right)_{u,v}.\end{aligned} \tag{20}$$

4.4. Error backpropagation in feature adaption layer. For the model, the error of the feature adaption layer will affect both the source domain and the target domain. Taking the target domain as an example, we calculated the gradient update error of the feature adaption layer.

In the feature adaption layer, the output is set as a_t^l. In the process of backpropagation, the error of this layer has two sources. One is the error gradient δ^{l+1} returned by the next layer over the network, and the other is the maximize mean discrepancies between the featurea_t^l and a_s^l. As for the feature adaption layer, the error can be expressed as:

$$\begin{aligned}J_t^l(W,b,x,y,\mathrm{MMD}) &= \frac{1}{2}\left\|a_t^l - y\right\|_2^2 + \gamma \mathrm{MMD}\left(a_t^l, a_s^l\right) \\ &\quad + \eta \sum_{k=1}^{K} \mathrm{MMD}\left(a_t^l | y_k, a_s^l | y_k\right).\end{aligned} \tag{21}$$

The gradient of W and b can be obtained as follows:

$$\begin{aligned}\frac{J_t^l(W,b,x,y,\mathrm{MMD})}{\partial W^l} &= \delta^l \left(a^{l-1}\right)^T + \gamma \frac{\partial \mathrm{MMD}\left(a_t^l, a_s^l\right)}{\partial W^l} \\ &\quad + \eta \sum_{k=1}^{K} \frac{\mathrm{MMD}\left(a_t^l | y_k, a_s^l | y_k\right)}{\partial W^l}, \\ \frac{J_t^l(W,b,x,y,\mathrm{MMD})}{\partial b^l} &= \delta^l + \gamma \frac{\partial \mathrm{MMD}\left(a_t^l, a_s^l\right)}{\partial b^l} \\ &\quad + \eta \sum_{k=1}^{K} \frac{\mathrm{MMD}\left(a_t^l | y_k, a_s^l | y_k\right)}{\partial b^l},\end{aligned} \tag{22}$$

Thus, the updated formula of weight and bias of each layer can be calculated as follows:

$$\begin{aligned}W^l &= W^l - \alpha \sum_{i=1}^{m} \left[a^{i,l-1} \cdot \delta^{i,l} + \gamma \frac{\partial \mathrm{MMD}\left(a_t^{i,l}, a_s^{i,l}\right)}{\partial W^{i,l}} \right. \\ &\quad \left. + \eta \sum_{k=1}^{K} \frac{\mathrm{MMD}\left(a_t^{i,l} | y_k, a_s^{i,l} | y_k\right)}{\partial W^{i,l}}\right], \\ b^l &= b^l - \alpha \sum_{i=1}^{m} \left[\sum_{u,v} \left(\delta^{i,l}\right)_{u,v} + \gamma \frac{\partial \mathrm{MMD}\left(a_t^{i,l}, a_s^{i,l}\right)}{\partial b^{i,l}} \right. \\ &\quad \left. + \eta \sum_{k=1}^{K} \frac{\mathrm{MMD}\left(a_t^{i,l} | y_k, a_s^{i,l} | y_k\right)}{\partial b^{i,l}}\right].\end{aligned} \tag{23}$$

After updating W and b of the feature adaption layer, it is also necessary to deduce the transfer of the errors of this layer to the previous layer. Referring to the backpropagation methods of the convolution layer, the error propagation variable of the feature adaption layer is defined as:

$$\begin{aligned}\delta^l &= \frac{\partial J(W,b,x,y,\mathrm{MMD})}{\partial z^l} \\ &= \left(\frac{\partial z^{l+1}}{\partial z^l}\right)^T \frac{\partial J(W,b,x,y,\mathrm{MMD})}{\partial z^{l+1}}.\end{aligned} \tag{24}$$

According to the matrix relation between two layers, we can figure out:

$$\begin{aligned}\frac{\partial z^{l+1}}{\partial z^l} &= \left(W^{l+1}\right)\mathrm{diag}\left(\sigma'\left(z^l\right)\right), \\ \frac{\partial J(W,b,x,y,\mathrm{MMD})}{\partial z^{l+1}} &= \delta^{l+1} + \gamma \frac{\partial \mathrm{MMD}\left(a_t^l, a_s^l\right)}{\partial z^{l+1}} \\ &\quad + \eta \sum_{k=1}^{K} \frac{\mathrm{MMD}\left(a_t^l | y_k, a_s^l | y_k\right)}{\partial z^{l+1}}.\end{aligned} \tag{25}$$

Thus, the recursive relation of the feature adaption layer can be obtained as follows:

$$\delta^l = \operatorname{diag}\left(\sigma'\left(z^l\right)\right)\left(W^{l+1}\right)^T \cdot \left[\delta^{l+1} + \gamma \frac{\partial \mathrm{MMD}\left(a_t^l, a_s^l\right)}{\partial z^{l+1}} + \eta \sum_{k=1}^{K} \frac{\mathrm{MMD}\left(a_t^l | y_k, a_s^l | y_k\right)}{\partial z^{l+1}}\right]. \tag{26}$$

5. Construction of testbed and application of the model

To verify the effectiveness of the method, an airborne fuel pump test platform was built to obtain the fault test data of the airborne fuel pump. Firstly, based on the FMECA analysis of the statistics data of the airborne fuel pump over recent years, the common fault modes of the airborne fuel pump were obtained to guide the test. Secondly, based on the analysis of the physical model of the airborne fuel pump, the fault injection test was carried out in the testbed to collect the fault data of the relevant fault modes of the airborne fuel pump. After obtaining the data, the bearing fault data of Western Reserve University combined with our experimental data was used to carry out transfer learning research.

5.1. FMECA of the airborne fuel pump. The airborne fuel pump is a core component of the fuel system, and the pump is responsible for the fuel supply and fuel transfer of the aircraft. The structure of the airborne fuel pump is shown in Figure 5. Searching the degradation law of the airborne fuel pump is the basis for the life prediction of the airborne fuel pump. The time sequence of the degradation data of the airborne fuel pump is the key to predict the trend of breaking down, then estimate the life span of the airborne fuel pump [36, 37].

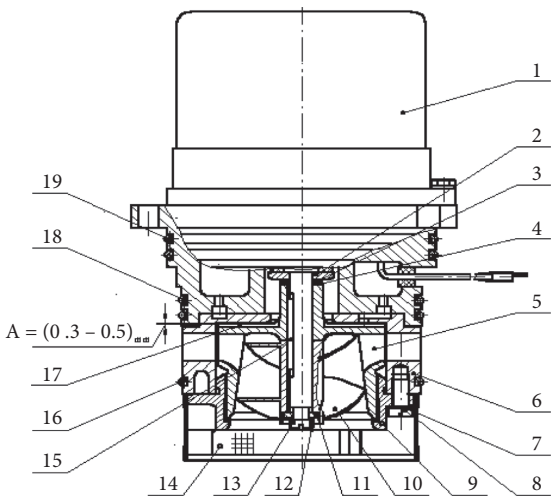
Through Failure Mode, Effects, and Criticality Analysis (FMECA) of the airborne fuel pump, six typical faults as blade damage, diffusion pipe damage, leakage, diffusion pipe and impeller rub, pump port and impeller rub, and bearing wear were selected, which is shown in Figure 6. Further analysis of the working principle and failure mechanism showed that when the fuel pump failure or performance declines, it will cause an abnormal vibration signal of the shell. However, in the military airfield, it is usually dismantled or returned to the factory for maintenance, without effective data monitoring and recording measures. Thus, it is difficult to quickly locate the fault, resulting in a reduction of the maintenance support level and waste of airborne equipment. Therefore, we considered selecting the vibration signal of the airborne fuel pump as the monitoring signal and carried out the time-frequency analysis and statistical characteristics analysis to extract the fault feature [38]. Our goal is to realize the intelligent and effective diagnosis of the airborne fuel pump.

5.2. Construction of airborne fuel pump testbed. A centrifugal AC electric pump provided by Nanjing Engineering Institute Centre is selected as the experiment object, as shown in Figure 7. This type of fuel pump is mainly used for the thermal subsystem and oil tanks. The fuel pump uses aviation fuel RP-3 as the working medium, whose temperature range from minus 60°C to 85°C. The other working parameters are shown in Table 1.

The experimental platform of the airborne fuel pump is shown in Figure 8. The platform mainly includes an oil storage tank, oil feeding tank, centrifugal test fuel pump, electric diaphragm pump, air-cooled radiator, pressure transducer, flow transducer, temperature transducer, liquid level transducer, data acquisition equipment, etc. In the main loop, the fuel pump pumps the oil from the oil feeding tank to the oil storage tank. For cycling, the oil in the storage tank returns to the feeding tank by gravity through the valve [39]. Through cycling, the working environment of the test pump is stable. In the second loop, an electric diaphragm pump is used to ensure the uniform distribution of particles in the impurity experiment. An air-cooled radiator is used to cool the oil and keep the oil temperature near room temperature. As shown in Figure 9, in the oil feeding tank, three vibration sensors are adopted to monitor the vibration of the airborne fuel pump, among which the vibration sensors are installed at three mutually perpendicular positions on the motor housing in the form of magnetic suction seats.

When testing, open the valve, fill the oil storage tank with fuel and connect the pump power supply, and make sure the pump continues to work. When the pump runs stably, collect the pump vibration signal, the outlet pressure signal, and the outlet flow signal. After the signal collection, close the power supply and the valve. Then, similar to the normal fuel pump, the other six typical fault signals are obtained by replacing different fault parts. The typical fault parts are shown in Figure 6. As shown in Table 2, vibration and pressure signals under normal state and 14 kinds of fault state were measured respectively in the experiment. Each group of data contained 4 channels, with a sampling frequency of 6000Hz and a sampling time of 5s for each channel.

5.3. Model application in the airborne fuel pump. In this paper, bearing fault data from Case Western Reserve University is selected as the source domain of transfer learning. Bearing faults in Case Western Reserve University are mainly caused by bearing wear, and the degree of bearing wear are 0.1778mm, 0.3556mm, 0.5334mm, and 0.7112mm. As for the target domain of airborne fuel pump, 1 impeller blade damage, diffusion tube damage, leakage, and bearing wear of 0.02mm were selected as the target domain for transfer learning. There are 4 types of ^{*3496} (vector number) ^{*246} (vector length) fault data available for the airborne fuel pump, and only 2^*246 data are selected for each type of fault. The remaining large amount of airborne fuel pump fault data are used as verification data for the transfer learning effect. In other words, the network trained by 2 sets of data was verified by 3494 sets of validation data to judge the diagnostic accuracy of the network. After the pre-processing, the small target samples combined with their labels are available for training.



1 Motor	2 Washer	3 Disc spring	4 Adjusting washer
5 Impeller	6 Diffusion tube	7 Washer	8 Cylinder head screw
9 Oil pump port	10 Axial flow impeller	11 Adjusting shim	12 Shim
13 Screws	14 Filters	15 Keys	16 Seals
17 Countersunk head screws	18 Seals	19 seals	

Figure 5: The structure of airborne fuel pump.

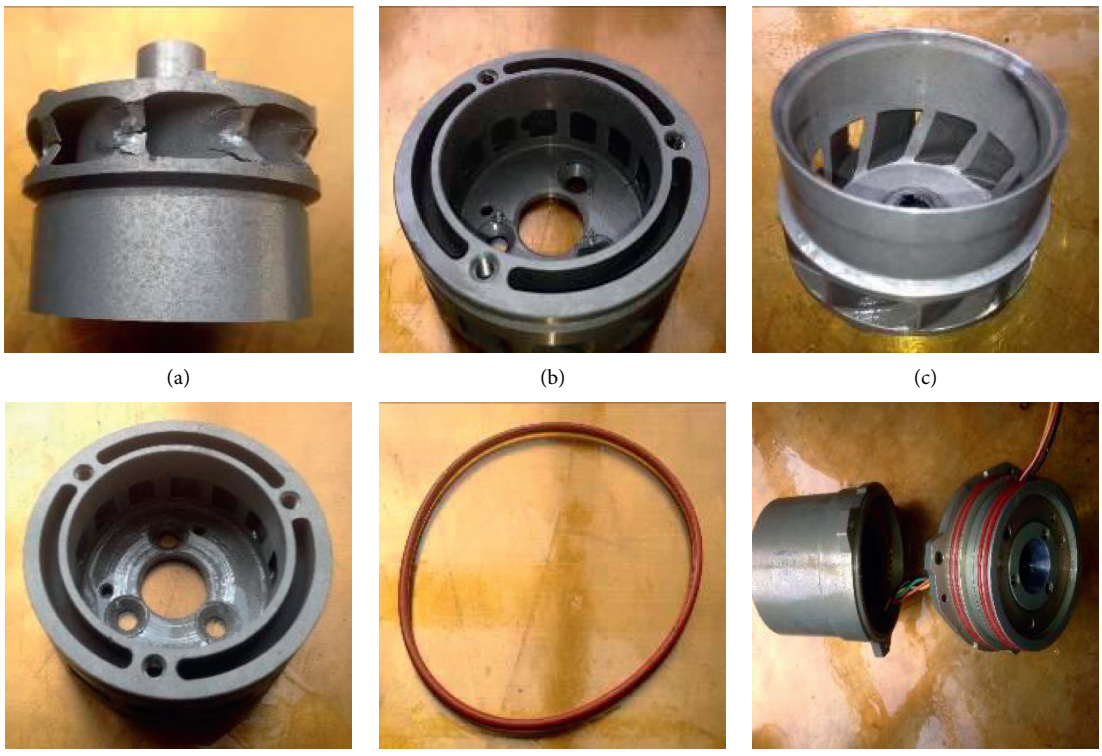


Figure 6: Typical faults of airborne fuel pump. (a) Blade damage; (b) Diffusion pipe damage; (c) Pump port and impeller rub; (d) Diffusion pipe and impeller rub; (e) Sealing ring aging; (f) Bearing wear.

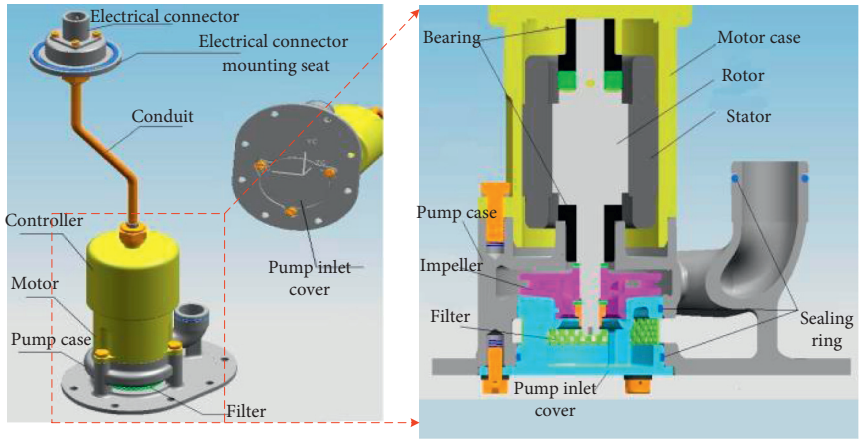


Figure 7: Structure diagram of a certain type centrifugal airborne fuel pump.

Table 1: Main working parameters of the airborne fuel pump.

Pump flow (L/h)	*Output pressure (Kpa)*	*Voltage (V)*	*Current (A)*	*Oil leakage (ml/min*
12000	≥73	115	≤5.5	0

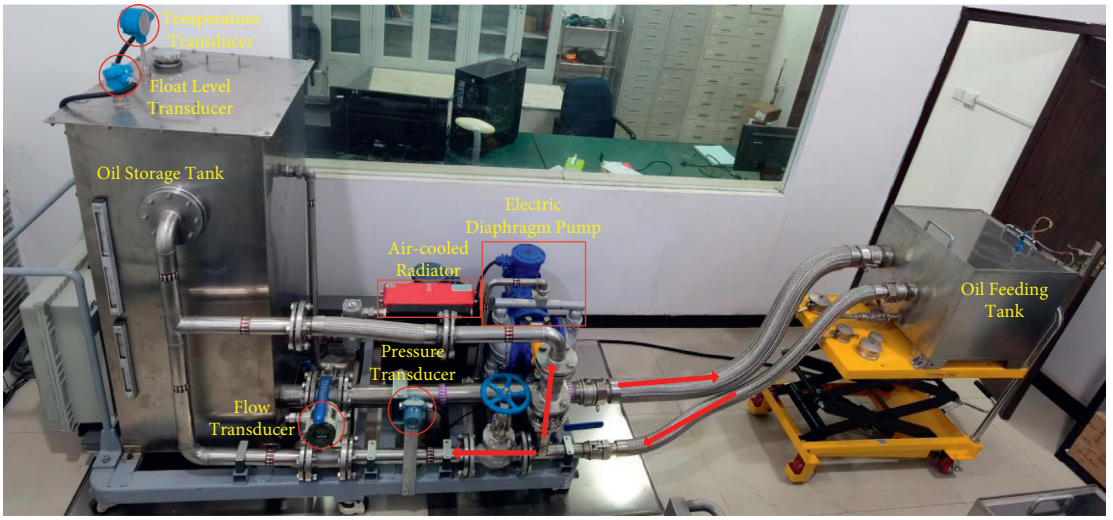


Figure 8: Testbed of the airborne fuel pump.

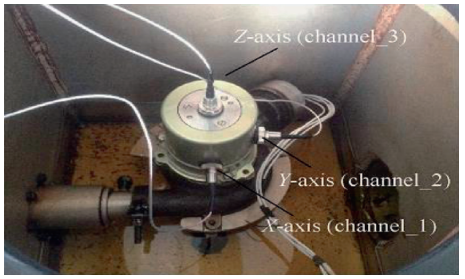


Figure 9: The installation of the airborne fuel pump in the testbed.

For the proposed model, there are hyper-parameters like learning rate, regularization parameter, cost function to be determined. In this research, we adopted randomized search to seek the optimized hyper-parameters. Nevertheless, the proposed model was complex that the process of hyper-parameters optimization cost too much time. Therefore, we sampled a smaller dataset to fed into the model. Although the accuracy of hyper-parameters decreased, the time cost reduced greatly. Through this way, the hyper-parameters of the model were determined [40].

To test the efficiency of the proposed model, firstly, fault diagnosis without transfer learning was carried out for small sample fault data of airborne fuel pump, and the confusion matrix of diagnosis results was shown in Figure 10. Then, the feature transfer model was used, bearing fault data from Case Western Reserve University was taken as the source domain, and the small sample fault data of airborne fuel pump was taken as the target domain. The results of the confusion matrix of the transfer diagnosis were shown in Figure 10. According to the results, the fault diagnosis

Table 2: Scheme of fault injection test.

NO.	Statement		Sample size (group)	Channel number	Time(second)	Frequency (Hz)
1		Normal	30	4	5	6000
2	foreign object damage	1 impeller blade	30	4	5	6000
3		2 impeller blades	30	4	5	6000
4		all impeller blades	30	4	5	6000
5		Diffusion tube	30	4	5	6000
6		all impeller blades & Diffusion tube	30	4	5	6000
7	scratch	back of impeller scrapes with diffusion tube	2	4	60	6000
8		edge of impeller inlet scrapes with oil pump mouth ring	2	4	60	6000
9		leakage	30	4	5	6000
10	bearing wear	Wear 0mm	15	4	5	6000
11		Wear 0.02mm	15	4	5	6000
12		Wear 0.05mm	15	4	5	6000
13		Wear 0.08mm	15	4	5	6000
14		Wear 0.10mm	15	4	5	6000
15		Wear 0.12mm	15	4	5	6000

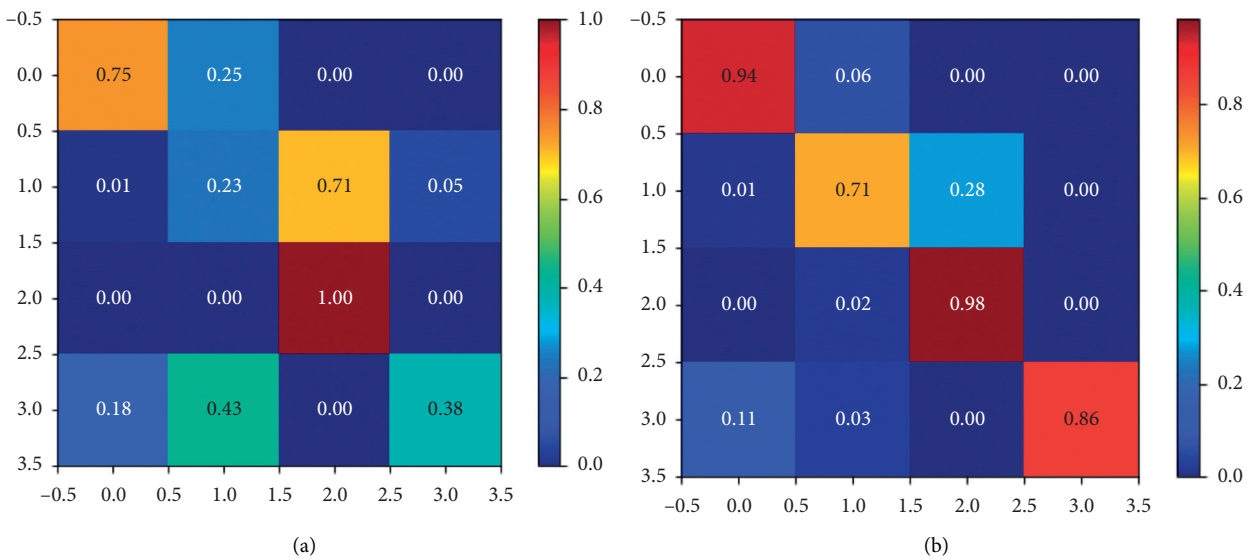


Figure 10: Confusion matrix of transfer learning results. (a) Confusion matrix without transfer learning; (b) Confusion matrix after transfer learning.

accuracy of each type of airborne fuel pump is improved by 28.25% on average through feature transfer learning.

Some other classical transfer learning algorithms such as ResNet-50, DANN, ADDA, JAN, MADA, CBST, CAN, CDAN+E, DM-ADA, 3CATN, ALDA are chosen as comparisons. The accuracy of diagnosis is given in Table 3. The results show that the diagnostic accuracy of the proposed model is remarkably higher than other algorithms. To further explore the capacity of the proposed model to filter the transferable data, some data filtering methods such as PCA, K-Means, DBSCAN, GMM, BIRCH are selected as comparisons. The accuracy of diagnosis is given in Table 4, the proposed model gets a higher score than models with other data filtering methods. The results show that the proposed model is more capable to filter available data in the source domain, which further proves that the proposed algorithm may prevent negative transfer to some extent.

5.4. Validation of feature transfer. To further verify the effectiveness of the model, this paper uses the method of feature visualization to show the learning effect of feature transfer. Since the features extracted from the network are in high-dimensional that cannot be directly visualized, the T-Distribution Stochastic Neighbour Embedding method is adopted to visualize the high-dimensional data.

T-distribution Stochastic Neighbour Embedding is often used to visualize high-dimensional data. The main advantage of T-SNE is the ability to preserve local structures. The

Table 3: Accuracy of diagnosis under different transfer learning approaches.

Model	Accuracy
Proposed model	87.3±0.5%
ResNet-50 (He et al., 2016)	63.2±0.8%
DANN (Ganin et al., 2016)	80.1±0.2%
ADDA (Tzeng et al., 2017)	53.8±1.5%
JAN (Long et al., 2017)	82.3±0.6%
MADA (Pei et al., 2018)	76.3±1.2%
CBST (Zou et al., 2018)	56.7±0.9%
CAN (Zhang et al., 2018)	75.1±0.5%
CDAN+E (Long et al., 2018b)	79.5±0.7%
DM-ADA (Xu et al., 2020)	58.7±1.1%
3CATN (Li et al., 2019)	83.2±0.6%
ALDA (Chen et al., 2020)	73.6±0.5%

Table 4: Accuracy of diagnosis with different data filtering methods.

Model	Accuracy
Proposed model (LLE+CNN+JDA)	87.3±0.5%
PCA+CNN+JDA	65.7±0.7%
K-Means+CNN+JDA	68.6±0.4%
DBSCAN+CNN+JDA	77.6±0.4%
GMM+CNN+JDA	85.2±0.6%
BIRCH+CNN+JDA	71.4±0.7%

Figure 11: Feature visualization by T-SNE. Red, blue, green, and purple represent sort 1-4 separately. (a) Feature visualization without transfer learning; (b) Feature visualization after transfer learning.

T-SNE algorithm models the distribution of the nearest neighbours of each data point. We model the high-dimensional space as a Gaussian distribution, while model the two-dimensional output space as a T-distribution. The goal of this process is to find the transformation that maps a higher-dimensional space to a two-dimensional space and to minimize the gap between these two distributions of all points [41].

For the network without transfer learning, the convolutional neural network for feature extraction was trained by 2 sets of data, and 3,494 sets of data were used as validation data. Feature extraction was carried out in the Convolutional Neural Network from validation data, and the extracted features were visualized using T-Distribution Stochastic Neighbour Embedding. The obtained results are shown in Figure 11(a). Similarly, for the model with transfer learning, the features extracted from the verification data were visualized using T-Distribution Stocking Neighbour Embedding, as shown in Figure 11(b). As can be seen from the figure, for networks without feature transfer, the boundaries of feature categories 1, 2, and 4 extracted from verification data are not clear, which is not conducive to the next step of feature classification and diagnosis. In the network with feature migration, the four categories of features extracted from the verification data have clear boundaries, which is easy to carry out classification and diagnosis. The effectiveness of the model for feature transfer is further proved from the feature visualization.

6. Conclusions

In this research, we proposed a feature transfer scenario that transfers knowledge from similar fields to enhance the accuracy of fault diagnosis with small sample. To reduces the

redundant information, data were filtered according to manifold consistency. Then, features were extracted based on CNN and feature transfer was conducted. For adequate fitness, the joint adaptation of conditional distribution and marginal distribution was used between the two domains. Minimum structural risk and MMD of adaptation were two indicators weighted for training the model. To test the efficiency of the model, we built an airborne fuel pump testbed, and contributed a new dataset that contained 15 categories of fault data, which serves as the small sample dataset in this research. Then the proposed model was applied in our experimental data. As a result, the fault diagnosis rate increases by 28.6% through our proposed model, which is more precise than other classical methods. The results of feature visualization further demonstrate that the features are more distinguished through the proposed method. All code and data are accessible on my GitHub.

Data Availability

Data and code of this research are accessible, please visit: https://github.com/ppqweasd/Diagnosis-for-High-reliability-Equipment-with-Small-Sample-Based-on-Transfer-Learning-A-General-Fra.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] Y. Wang, Z. Pan, X. Yuan, C. Yang, and W. Gui, "A novel deep learning based fault diagnosis approach for chemical process with extended deep belief network," *ISA Transactions*, vol. 96, pp. 457–467, 2020.

[2] N. Md Nor, C. R. Che Hassan, and M. A. Hussain, "A review of data-driven fault detection and diagnosis methods: applications in chemical process systems," *Reviews in Chemical Engineering*, vol. 36, no. 4, pp. 513–553, 2020.

[3] Y. Wang, H. Yang, X. Yuan, Y. A. W. Shardt, C. Yang, and W. Gui, "Deep learning for fault-relevant feature extraction and fault classification with stacked supervised auto-encoder," *Journal of Process Control*, vol. 92, pp. 79–89, 2020.

[4] J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu, "Deep learning for smart manufacturing: methods and applications," *Journal of Manufacturing Systems*, vol. 48, pp. 144–156, 2018.

[5] T. Kinoshita, K. Fujiwara, Y. Sumi, M. Matsuo, M. Kano, and H. Kadotani, "Development of spindle detection algorithm by wavelet synchrosqueezed transform and random under sampling," *Sleep Medicine*, vol. 64, p. S121, 2019.

[6] P. Soltanzadeh and M. Hashemzadeh, "RCSMOTE: range-Controlled synthetic minority over-sampling technique for handling the class imbalance problem," *Information Sciences*, vol. 542, pp. 92–111, 2021.

[7] N.-T. Tran, V.-H. Tran, N.-B. Nguyen, T.-K. Nguyen, and N.-M. Cheung, "On data augmentation for gan training," *IEEE Transactions on Image Processing*, vol. 30, pp. 1882–1897, 2021.

[8] X. Zhou, X. Liu, G. Lan, and J. Wu, "Federated conditional generative adversarial nets imputation method for air quality missing data," *Knowledge-Based Systems*, vol. 228, 2021.

[9] T. Lee and S. Yoo, "Augmenting few-shot learning with supervised contrastive learning," *Ieee Access*, vol. 9, pp. 61466–61474, 2021.

[10] R.-Q. Wang, X.-Y. Zhang, and C.-L. Liu, *Meta-Prototypical Learning for Domain-Agnostic Few-Shot Recognition*, IEEE Trans. Neural Netw. Learn. Syst., 2021.

[11] F. Zhuang, Z. Qi, K. Duan, D. Xi, and Q. He, "A comprehensive survey on transfer learning," *Proceedings of the IEEE*, vol. 109, no. 99, pp. 1–34, 2020.

[12] Z. P. Yu, J. H. Zhao, Y. C. Wang, L. L. He, and S. N. Wang, "Surface EMG-based instantaneous hand gesture recognition using convolutional neural network with the transfer learning method," *Sensors*, vol. 21, no. 7, p. 21, 2021.

[13] S. Smetanin and M. Komarov, "Deep transfer learning baselines for sentiment analysis in Russian," *Information Processing & Management*, vol. 58, no. 3, p. 19, 2021.

[14] A. Langevin, T. Cody, S. Adams, and P. Beling, "Generative adversarial networks for data augmentation and transfer in credit card fraud detection," *Journal of the Operational Research Society*, p. 28, 2021.

[15] N. Wu, F. Liu, F. J. Meng, M. Li, C. Zhang, and Y. He, "Rapid and accurate varieties classification of different crop seeds under sample-limited condition based on hyperspectral imaging and deep transfer learning," *Frontiers in Bioengineering and Biotechnology*, vol. 9, p. 14, 2021.

[16] F. Liu, J. Lu, B. Han, G. Niu, G. Zhang, and M. Sugiyama, "Butterfly: one-step approach towards wildly unsupervised domain adaptation," 2019.

[17] X. Chen, X. Lan, F. Sun, and N. Zheng, "A boundary based out-of-distribution classifier for generalized zero-shot learning," 2020.

[18] T. Teshima, I. Sato, and M. Sugiyama, "Few-shot domain adaptation by causal mechanism transfer," 2020.

[19] Y. L. Wang, D. Z. Wu, and X. F. Yuan, "LDA-based deep transfer learning for fault diagnosis in industrial chemical processes," *Computers & Chemical Engineering*, vol. 140, pp. 13-14, 2020.

[20] V. Singh and N. K. Verma, "mRMR-DNN with transfer learning for IntelligentFault diagnosis of rotating machines," 2019.

[21] Y. Deng, D. Huang, S. Du, G. Li, C. Zhao, and J. Lv, "A double-layer attention based adversarial network for partial transfer learning in machinery fault diagnosis," *Computers in Industry*, vol. 127, Article ID 103399, 2021.

[22] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[23] R. Yan, F. Shen, C. Sun, and X. Chen, "Knowledge transfer for rotary machine fault diagnosis," *IEEE Sensors Journal*, vol. 20, no. 15, p. 1, 2019.

[24] Z. Yue, L. Meng, L. Liangzhen, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-IID data arXiv," vol. 13, p. 13, 2018.

[25] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: a geometric framework for learning from labeled and unlabeled examples," *Journal of Machine Learning Research*, vol. 7, pp. 2399–2434, 2006.

[26] Juba and Brendan, "Estimating relatedness via data compression," in *Proceedings of the 23rd international conference on Machine learning*, pp. 441–448, USA, June 2006.

[27] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[28] M. Rezaei-Ravari, M. Eftekhari, and F. Saberi-Movahed, "Regularizing extreme learning machine by dual locally linear embedding manifold learning for training multi-label neural network classifiers," *Engineering Applications of Artificial Intelligence*, vol. 97, p. 15, 2021.

[29] J. Shawe-Taylor, P. L. Bartlett, R. C. Williamson, and M. Anthony, "Structural risk minimization over data-dependent hierarchies," *IEEE Transactions on Information Theory*, vol. 44, no. 5, pp. 1926–1940, 1998.

[30] J. Liu, M. Bai, N. Jiang, and D. Yu, "Structural risk minimization of rough set-based classifier," *Soft Computing*, vol. 24, no. 3, pp. 2049–2066, 2020.

[31] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer feature learning with joint distribution adaptation," *Proceedings*, in *Proceedings of the IEEE International Conference on Computer Vision*, Sydney, NSW, Australia, December 2013.

[32] A. Smola, A. Gretton, L. Song, and B. Schölkopf, *A Hilbert Space Embedding for Distributions*, pp. 13–31, Springer, Berlin, Germany, 2007.

[33] A. Gretton, K. M. Borgwardt, M. Rasch, B. Schlkopf, and A. J. Smola, "A kernel two-sample test," *Journal of Machine Learning Research*, vol. 13, pp. 723–773, 2012.

[34] M. Long, J. Wang, G. Ding, S. J. Pan, and P. S. Yu, "Adaptation regularization: a general framework for transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 5, pp. 1076–1089, 2014.

[35] M. Long, J. Wang, G. Ding, D. Shen, and Q. Yang, "Transfer learning with graph Co-regularization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 7, pp. 1805–1818, 2014.

[36] X. Jiao, B. Jing, X. Qiang, Q. Liu, J. Li, and W. Zhou, "Fault diagnosis of airborne fuel pump and experimental platform research," vol. 36, no. 01, pp. 120–128, 2017.

[37] J. X. Pan, B. Jing, Z. Li, S. Wang, and X. X. Jiao, "Modeling and experimental design of electrical stress accelerated degradation of airborne fuel pump," *Journal of Electronic Measurement and Instrument*, vol. 33, no. 11, pp. 50–56, 2019.

[38] X. Jiao, B. Jing, Y. Huang, J. Li, and G. Xu, "Research on fault diagnosis of airborne fuel pump based on EMD and probabilistic neural networks," *Microelectronics Reliability*, vol. 75, no. aug, pp. 296–308, 2017.

[39] J. X. Pan, B. Jing, X. X. Jiao, and S. L. Wang, "Analysis and application of grey wolf optimizer-long short-term memory," *IEEE Access*, vol. 8, pp. 121460–121468, 2020.

[40] M. Abd Elaziz, A. Dahou, L. Abualigah et al., "Advanced metaheuristic optimization techniques in applications of deep neural networks: a review," *Neural Computing & Applications*, vol. 33, no. 21, pp. 14079–14099, 2021.

[41] V. D. M. Laurens and G. Hinton, "Visualizing Data using t-SNE," *Journal of Machine Learning Research*, vol. 9, no. 2605, pp. 2579–2605, 2008.