






Transfer Representation Learning With TSK Fuzzy System

Peng Xu, Zhaohong Deng , Senior Member, IEEE, Jun Wang , Member, IEEE, Qun Zhang ,
Kup-Sze Choi , Member, IEEE, and Shitong Wang 

Abstract—Transfer learning can address the learning tasks of unlabeled data in the target domain by leveraging plenty of labeled data from a different but related source domain. A core issue in transfer learning is to learn a shared feature space where the distributions of the data from the two domains are matched. This learning process can be named as transfer representation learning (TRL). Feature transformation methods are crucial to ensure the success of TRL. The most commonly used feature transformation method in TRL is kernel-based nonlinear mapping to the high-dimensional space, followed by linear dimensionality reduction. But the kernel functions are lack of interpretability, and it is difficult to select kernel functions. To this end, this article proposes a more intuitive and interpretable method, called TRL with TSK-FS (TRL-TSK-FS), by combining TSK fuzzy system (TSK-FS) with transfer learning. Specifically, TRL-TSK-FS realizes TRL from two aspects. On one hand, the data in the source and target domains are transformed into the fuzzy feature space where the distribution distance of the data between the two domains is minimized. On the other hand, discriminant information and geometric properties of the data are preserved by linear discriminant analysis and principal component analysis. A further advantage is that nonlinear transformation is realized in the proposed method by constructing fuzzy mapping with the antecedent part of the TSK-FS instead of kernel functions, which are difficult to be selected. Extensive experiments are conducted on text and image datasets to demonstrate the superiority of the proposed method.

Index Terms—Fuzzy feature space, transfer representation learning (TRL), TSK fuzzy system (TSK-FS), unsupervised domain adaptation.

I. INTRODUCTION

MACHINE learning algorithms often assume that the samples are independent and identically distributed and that a classifier can be trained with abundant labeled data. However, the algorithms are often confronted with two problems when they are applied to applications, such as natural language processing [1], computer vision [2], and health informatics [3]. On one hand, labeled data are not always fully provided. On the other hand, the training data and test data collected may be of different distributions in real-world applications. To develop an effective technique to solve the two problems, transfer learning has been widely studied in recent years. It can leverage the training data from the source domain to assist in tackling the learning tasks in the target domain. For example, the sentiment classification of movie reviews on a new movie website, where the labeled movie reviews are rare, can leverage the abundant labeled data of food reviews from another website to train the classifier.

Domain adaptation is an important paradigm in transfer learning research, which assumes that the data from both the source domain and the target domain, albeit of different distributions, concern the same task [4]. According to the availability of the labeled data in the target domain, domain adaptation can be categorized as semisupervised domain adaptation and unsupervised domain adaptation. The former has a small amount of labeled data in the target domain, whereas the latter has no labeled data in the target domain [5]. This article focuses on unsupervised domain adaptation, which has fewer restrictions on the data. Current research works on transfer learning mainly focus on feature-based transfer and model-based transfer. There are also studies exploring sample-based transfer [6] or the integration of these paradigms [7], [8]. The two mainstream paradigms are introduced below, and their drawbacks are summarized accordingly.

Model-based transfer learning refers to transfer using the parametric relationship between the source domain and target domain. Model-based transfer learning methods often directly yield a classifier for the target domain. Typical characteristics of these algorithms are the parametric correlation hypothesis [9]–[11] or the parametric sharing hypothesis [12], [13]. The

Manuscript received September 27, 2018; revised July 18, 2019; accepted November 28, 2019. Date of publication December 9, 2019; date of current version March 1, 2021. This work was supported in part by the NSFC under Grant 61772239, in part by the Jiangnan University State Key Laboratory of Food Science and Technology Free Exploration Project under Grant SKLF-ZZB-201901, in part by the National First-Class Discipline Program of Light Industry Technology and Engineering under Grant LITE2018-02 and Grant LITE2018-03, in part by the Six Talent Peaks Project in Jiangsu Province under Grant XYDXX-056, in part by the General Project of the National Social Science Fund of China under Grant 19BTQ030, in part by the Jiangsu Province Natural Science Fund under Grant BK20181339. (Corresponding author: Zhaohong Deng.)

P. Xu and S. Wang are with the School of Digital Media and the Jiangsu Key Laboratory of Digital Design and Software Technology, Jiangnan University, Wuxi 214122, China (e-mail: pengxujnu@163.com; wxwangst@aliyun.com).

Z. Deng is with the School of Digital Media, the Jiangsu Key Laboratory of Digital Design and Software Technology, and the State Key Laboratory of Food Science and Technology, Jiangnan University, Wuxi 214122, China (e-mail: dengzhaohong@jiangnan.edu.cn).

J. Wang is with the Shanghai Institute for Advanced Communication and Data Science, School of Communication and Information Engineering, Shanghai University, Shanghai 20444, China (e-mail: wangjun_sytu@hotmail.com).

Q. Zhang is with the Library, Jiangnan University, Wuxi 214122, China (e-mail: 961044284@qq.com).

K.-S. Choi is with the Centre for Smart Health, Hong Kong Polytechnic University, Hong Kong (e-mail: thomask.s.choi@polyu.edu.hk).

Color versions of one or more of the figures in this article are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TFUZZ.2019.2958299

core idea of these methods is to introduce parametric relationship into the training process of the classifier for the target domain. In [9]–[11], based on parametric correlation hypothesis, an auxiliary classifier on the source domain is used to improve classifier training on the target domain. In [12] and [13], based on parametric sharing hypothesis, a distribution distance regularization term is added to the objective function of the classifier.

Feature-based transfer learning algorithms assume that there exists a shared feature space between the source domain and the target domain and that the distributions of the data from two domains are consistent in the shared feature space. The core idea of these methods is to learn such a feature space for the data from the two domains, and then arbitrary classifiers are applied to the data in this feature space. To achieve nonlinear feature transformation when learning the shared feature space, kernel functions are the most commonly used technique to transform the raw data into the high-dimensional space. Linear dimensionality reduction is adopted to map the data in the high-dimensional space to the low-dimensional shared feature space [8], [14]–[16]. There are mainly two ways to construct the shared feature space, and they are reviewed in Section II.

Model-based and feature-based transfer learning methods have several drawbacks. On one hand, the performance of most model-based transfer learning algorithms heavily depends on the form of the selected classifier. On the other hand, the discriminant information and geometric properties of the data in feature-based transfer learning can be impaired during feature transformation. There have been some work trying to alleviate this problem [2], [17]. Besides, the kernel functions lack interpretability, and it is thus difficult to select an appropriate kernel function to achieve nonlinear transformation.

As a more intuitive and interpretable modeling method, TSK fuzzy system (TSK-FS) has been applied to various applications [18]–[20]. It is a kind of data-driven system composed of several IF–THEN fuzzy rules that offer high interpretability. Some transfer fuzzy systems have been proposed to provide better interpretability and deal with the uncertainty for transfer learning [21]. Based on TSK-FS, a model-based transfer learning method and its enhanced version are proposed in [22] and [23], respectively. They assume that the consequent parameters of TSK-FS between the source and target domains are correlated. A model-based transfer learning method with parametric sharing hypothesis for consequent parameters of TSK-FS is developed in [24]. In [25], a set of fuzzy rules of TSK-FS is first constructed on the source domain, and then the feature space of the target domain is modified by learning a mapping to match the existing rules. As an extended work of [25], Zuo *et al.* [26] proposes label space adaptation by learning another mapping in addition to input space adaptation. To improve the performance of transfer learning based on fuzzy systems, semisupervised learning and active learning are also introduced to fuzzy transfer learning in [27] and [28]. Besides, the inherent phenomenon of information granularity in transfer learning is taken into account by introducing granular computing technique to fuzzy transfer [29]. Although these methods have achieved promising

performance, they are all model-based transfer learning methods whose performance depends heavily on the learning ability of the specific fuzzy systems.

To overcome the aforementioned drawbacks of the existing transfer learning algorithms, a feature-based transfer learning method based on TSK-FS, i.e., transfer representation learning with TSK-FS (TRL-TSK-FS) is proposed in this article. TRL-TSK-FS treats a multioutput TSK-FS as the feature mapping and constructs a shared feature space for the data from the source and target domains. The proposed method realizes transfer representation learning from two aspects. First, the distribution distance of the data from the two domains is minimized in the fuzzy feature space. Second, the discriminant information and geometric properties are preserved by the forms of linear discriminant analysis (LDA) and principal component analysis (PCA).

TRL-TSK-FS has the following advantages. First, unlike the existing model-based transfer fuzzy systems, TRL-TSK-FS is a feature-based method whose classifier selection is therefore more flexible, and the performance is not affected by the specific classifier. Second, TRL-TSK-FS leverages the discriminant information and preserves the geometric properties of the data with the forms of LDA and PCA during feature transformation. Third and most importantly, TRL-TSK-FS considers multioutput TSK-FS as a feature transformation method, which realizes nonlinear transformations and linear dimensionality reduction simultaneously. Unlike the classical transfer representation learning methods, which adopted kernel methods to realize nonlinear transformation [8], [14], [16], [17], the proposed method realizes nonlinear transformation by constructing the fuzzy mapping with the antecedent part of TSK-FS. The fuzzy mapping not only avoids kernel function selection, but also preserves more information in the original data. Besides, the consequent part of TSK-FS is regarded as the linear dimensionality reduction, which maps the data from high-dimensional space generated by the antecedents to the low-dimensional shared feature space.

The main contributions of this article are summarized as follows.

- 1) This article introduces the TSK-FS into the feature-based transfer learning and proposes the novel TRL-TSK-FS method to learn a shared feature space in which the distribution distance and the information loss of the data are minimized simultaneously.
- 2) A novel method for calculating the antecedent parameters of TSK-FS is proposed, which is based on the deterministic clustering algorithm Var-Part, and avoids the initialization sensitivity problem in the traditional clustering-based TSK-FS modeling methods.
- 3) Extensive experiments are conducted on the image dataset Office-Caltech and the text dataset NG20. The experimental results clearly demonstrate the effectiveness and superiority of the proposed method.

The remainder of this article is organized as follows. Some classical transfer representation learning methods are reviewed in Section II, followed by the fundamentals of the proposed method. Section III gives the details of the proposed method,

and Section IV presents and analyzes the experimental results. Section V concludes this article.

II. RELATED WORK

The existing feature-based transfer learning algorithms are first reviewed, followed by the fundamentals of the proposed methods, which includes TSK-FS and maximum mean discrepancy (MMD).

A. Transfer Representation Learning

The feature-based transfer learning is also known as transfer representation learning. There are mainly two ways to construct the shared subspace in transfer representation learning, i.e., hidden subspace construction and subspace alignment (SA). Pan *et al.* [30] formulated the minimization of the distribution distance between the source and target domains in the new shared subspace as the problem of solving the kernel matrix. A nonlinear mapping was obtained implicitly by solving the kernel matrix with semidefinite programming. Linear dimensional reduction was then conducted on the data mapped with the kernel matrix, and the hidden subspace is obtained. However, semidefinite programming has high computational complexity and is not suitable for large-scale data. To improve the work presented in [30], the transfer component analysis (TCA) proposed in [14] directly minimized the distribution distance of the data of the two domains in the hidden subspace constructed by kernel function mapping and linear dimensional reduction. TCA formulated the optimization problems by using generalized eigenvalue decomposition. To extend TCA, Long *et al.* [16] proposed the joint distribution adaptation (JDA), which minimized the distribution distance of the two domains, not only in marginal distribution but also in conditional distribution. By integrating sample-based transfer and feature-based transfer, transfer joint matching (TJM) was proposed in [8]. To preserve the geometric properties of the original data in the process of feature transformation, scatter component analysis (SCA) was proposed in [17].

There are two problems with the abovementioned algorithms when constructing the hidden subspace: 1) A hidden subspace may not exist, and 2) it is difficult to select an appropriate kernel function to achieve nonlinear transformation. To deal with the first problem, SA has been widely used in transfer representation learning. In the SA algorithm proposed in [31], dimension reduction for the source and target domains was first conducted using PCA, and the linear mapping from the source domain to the target domain is then solved. Aljundi *et al.* [32] introduced the kernel technique into SA to realize nonlinear transformation. For the second problem, the multiple kernel learning method was introduced into transfer learning [33]. In this article, the proposed method TRL-TSK-FS will explicitly construct a nonlinear transformation through fuzzy mapping to solve this problem.

B. TSK Fuzzy System

TSK-FS [34] is an intelligent model based on fuzzy logic [35]. Similar to machine learning methods, it learns the model

parameters by data-driven approach [36]. TSK-FS has good interpretability and can be formulated with “IF-THEN” rules as follows:

$$\begin{aligned} \text{IF: } & x_1 \text{ is } A_1^k \wedge x_2 \text{ is } A_2^k \wedge \cdots \wedge x_d \text{ is } A_d^k \\ \text{THEN: } & f^k(\mathbf{x}) = p_0^k + p_1^k x_1 + \cdots + p_d^k x_d \end{aligned} \quad (1)$$

where $k = 1, 2, \dots, K$, K is the number of rules, $\mathbf{x} \in R^{d \times 1}$, d is the number of dimensions of the samples, $f^k(\mathbf{x})$ represents the output of the k th rule of TSK-FS, and A_i^k represents a fuzzy set. Unlike the crisp set where the membership can only be 0 or 1, the membership in fuzzy set can be any values between 0 and 1. The membership can be calculated using membership functions, and the definition of the membership functions is the core issue of TSK-FS.

Depending on application scenarios, different forms of membership functions can be defined. In the absence of domain knowledge, a commonly used fuzzy membership function is the Gaussian function, as given in (2) [37]. Each fuzzy set has a corresponding membership function, and the parameters c_i^k, δ_i^k in the membership function are called antecedent parameters. Fuzzy c-means (FCM) is often used to calculate the antecedent parameters [24]

$$\mu_{A_i^k}(x_i) = \exp\left(-(x_i - c_i^k)/2\delta_i^k\right)^2. \quad (2)$$

With known antecedent parameters, the membership value of each feature of the corresponding fuzzy set A_i^k can be calculated by (2). If multiplication is used as conjunction operator, the firing level of the k th rule of each sample can be calculated by (3). The normalized form of (3) is given in (4). The output of the TSK-FS is the weighted average of $f^k(\mathbf{x})$, as shown in (5).

$$\mu^k(\mathbf{x}) = \prod_{i=1}^d \mu_{A_i^k}(x_i) \quad (3)$$

$$\tilde{\mu}^k(\mathbf{x}) = \mu^k(\mathbf{x}) / \sum_{k'=1}^K \mu^{k'}(\mathbf{x}) \quad (4)$$

$$f(\mathbf{x}) = \sum_{k=1}^K \tilde{\mu}^k(\mathbf{x}) f^k(\mathbf{x}) \quad (5)$$

Once the antecedent parameters are obtained, the output of TSK-FS in (5) can be expressed as the form of linear regression as follows [38]:

$$y = f(\mathbf{x}) = \mathbf{p}_g^T \mathbf{x}_g \quad (6)$$

where

$$\mathbf{x}_e = [1, \mathbf{x}^T]^T \in R^{(d+1) \times 1} \quad (7a)$$

$$\tilde{\mathbf{x}}^k = \tilde{\mu}^k(\mathbf{x}) \mathbf{x}_e \in R^{(d+1) \times 1} \quad (7b)$$

$$\mathbf{x}_g = [(\tilde{\mathbf{x}}^1)^T, (\tilde{\mathbf{x}}^2)^T, \dots, (\tilde{\mathbf{x}}^K)^T]^T \in R^{K(d+1) \times 1} \quad (7c)$$

$$\mathbf{p}^k = [p_0^k, p_1^k, \dots, p_d^k]^T \in R^{(d+1) \times 1} \quad (7d)$$

and

$$\mathbf{p}_g = [(\mathbf{p}^1)^T, (\mathbf{p}^2)^T, \dots, (\mathbf{p}^K)^T]^T \in R^{K(d+1) \times 1}. \quad (7e)$$

Here, \mathbf{x}_g represents the feature vector through the fuzzy mapping of the antecedent part of TSK-FS, and \mathbf{p}_g represents the consequent parameters of the TSK-FS. Equation (6) is a linear model, and \mathbf{p}_g can be solved by least square method with the known \mathbf{x}_g .

C. Maximum Mean Discrepancy

Transfer representation learning assumes that the distribution distance between the source and target domains can be minimized in the new feature space. MMD is a commonly used distribution distance measure in transfer learning [14], [39]. MMD is a two-sample statistical test method. According to the distribution distance of the observed data $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^m$ and $\mathbf{Y} = \{\mathbf{y}_j\}_{j=1}^n$ of distribution p and q , respectively, and MMD can reject or accept the null hypothesis $p = q$. In practice, the empirical estimation of MMD can be formulated as follows [40]:

$$\text{MMD}^2(\mathbf{X}, \mathbf{Y}) = \left\| \frac{1}{m} \sum_{i=1}^m \phi(\mathbf{x}_i) - \frac{1}{n} \sum_{j=1}^n \phi(\mathbf{y}_j) \right\|_{\mathcal{H}}^2 \quad (8)$$

where ϕ is the feature mapping, and $\|\cdot\|_{\mathcal{H}}$ represents the reproducing kernel Hilbert space. Minimizing (8) means that the distribution distance between the two domains under the mapping ϕ is to be minimized. The goal of the transfer learning is thus to solve for ϕ .

For unsupervised domain adaptation, denote the data and the labels in the source domain as $\mathbf{X}_S = \{\mathbf{x}_{s_i}\}_{i=1}^{n_s}$ and $\mathbf{Y}_S = \{y_{s_i}\}_{i=1}^{n_s}$, and data with pseudo-labels $\mathbf{X}_T = \{\mathbf{x}_{t_j}\}_{j=1}^{n_t}$, $\hat{\mathbf{Y}}_T = \{\hat{y}_{t_j}\}_{j=1}^{n_t}$ in the target domain. Let $P_S(\mathbf{X}_S)$ and $P_T(\mathbf{X}_T)$ represent the marginal distributions, and $Q_S(\mathbf{Y}_S|\mathbf{X}_S)$ and $Q_T(\mathbf{Y}_T|\mathbf{X}_T)$ represent the conditional distributions of the two domains. To simultaneously adapt to the marginal distributions and conditional distributions of the two domains, the MMD can be defined as follows: [16]:

$$\text{MMD}_P^2(\mathbf{X}_S, \mathbf{X}_T) = \left\| \frac{1}{n_s} \sum_{i=1}^{n_s} \phi(\mathbf{x}_{s_i}) - \frac{1}{n_t} \sum_{j=1}^{n_t} \phi(\mathbf{x}_{t_j}) \right\|_{\mathcal{H}}^2 \quad (9a)$$

$$\begin{aligned} & \text{MMD}_Q^2(\mathbf{X}_S, \mathbf{X}_T) \\ &= \sum_{c=1}^C \left\| \frac{1}{n_s^{(c)}} \sum_{y_{s_i}=c} \phi(\mathbf{x}_{s_i}) - \frac{1}{n_t^{(c)}} \sum_{\hat{y}_{t_j}=c} \phi(\mathbf{x}_{t_j}) \right\|_{\mathcal{H}}^2 \end{aligned} \quad (9b)$$

where C is the number the classes, $n_s^{(c)}$ represents the number of examples belonging to the c th class in the source domain, and $n_t^{(c)}$ represents the number of examples belonging to the c th class in the target domain. The initial pseudo-labels for the target domain are obtained by the classifier trained on the source domain. Iterative strategy is adopted to improve the accuracy of the pseudo-labels, where in each iteration, the joint distributions are matched and the classifier is then trained again to label the data in the target domain.

III. TRANSFER REPRESENTATION LEARNING WITH TSK-FS

A. Overview of TRL-TSK-FS

Shared feature space construction is fundamental in transfer representation learning. During the procedure of feature transformation, the constraints for transfer learning are added into the objective function. There are two crucial constraints for transfer representation learning. One is distribution matching where the distribution distance of the two domains is minimized. The other is the preservation of the discriminant information and geometric properties. Based on aforementioned discussion, the problem can be expressed as follows:

$$\min_{\phi} \text{Distance}(\mathbf{X}_S, \mathbf{X}_T|\phi) + \text{Info_loss}(\mathbf{X}_S, \mathbf{X}_T|\phi). \quad (10)$$

The first term represents distribution matching under the mapping ϕ . The purpose of the second term is to minimize the information loss that contains discriminant information and geometric properties. ϕ can be learned by minimizing (10). In the existing methods, the most common form of ϕ is kernel mapping, followed by linear dimensionality reduction. The form of ϕ in TRL-TSK-FS is the TSK-FS.

The rationales of the proposed TRL-TSK-FS are described through five parts as follows. The construction method of the shared feature space through TSK-FS is described in detail in Section III-B. The two constraints for transfer learning are represented in Sections III-C and III-D, respectively. The optimization of the overall objective function and the analysis of the computational complexity are given in Sections III-E and III-F.

B. Shared Feature Space Construction

In general, there are two steps to construct the feature space, i.e., nonlinear transformation and linear dimensionality reduction. The proposed method realizes nonlinear transformation by the antecedent part of the multioutput TSK-FS and linear dimensionality reduction by the consequent part of the multioutput TSK-FS. The flowchart of the transformation procedure is illustrated in Fig. 1.

1) *Fuzzy Feature Space Based on TSK-FS*: Consider a multioutput TSK-FS as feature transformation ϕ , for an example \mathbf{x}_{s_i} in the source domain or an example \mathbf{x}_{t_i} in the target domain, the transformed data \mathbf{g}_{s_i} and \mathbf{g}_{t_i} through the antecedent part based on (7c) can be represented as follows:

$$\mathbf{g}_{s_i} = [(\tilde{\mathbf{x}}_{s_i}^1)^T, (\tilde{\mathbf{x}}_{s_i}^2)^T, \dots, (\tilde{\mathbf{x}}_{s_i}^K)^T]^T \in R^{K(d+1) \times 1} \quad (11a)$$

$$\mathbf{g}_{t_i} = [(\tilde{\mathbf{x}}_{t_i}^1)^T, (\tilde{\mathbf{x}}_{t_i}^2)^T, \dots, (\tilde{\mathbf{x}}_{t_i}^K)^T]^T \in R^{K(d+1) \times 1} \quad (11b)$$

$$\mathbf{G}_S = [\mathbf{g}_{s_1}, \mathbf{g}_{s_2}, \dots, \mathbf{g}_{s_{n_s}}] \in R^{K(d+1) \times n_s} \quad (11c)$$

$$\mathbf{G}_T = [\mathbf{g}_{t_1}, \mathbf{g}_{t_2}, \dots, \mathbf{g}_{t_{n_t}}] \in R^{K(d+1) \times n_t} \quad (11d)$$

$$\mathbf{P} = [\mathbf{p}_g^1, \mathbf{p}_g^2, \dots, \mathbf{p}_g^m] \in R^{K(d+1) \times m} \quad (11e)$$

where \mathbf{G}_S in (11c) is the concatenated data of \mathbf{g}_{s_i} for all examples in the source domain, and \mathbf{G}_T in (11d) can be obtained in a similar way. The main difference between the multioutput TSK-FS and the single-output TSK-FS in (6) is that the former involves multiple-group consequent parameters. \mathbf{P} in (11e) is

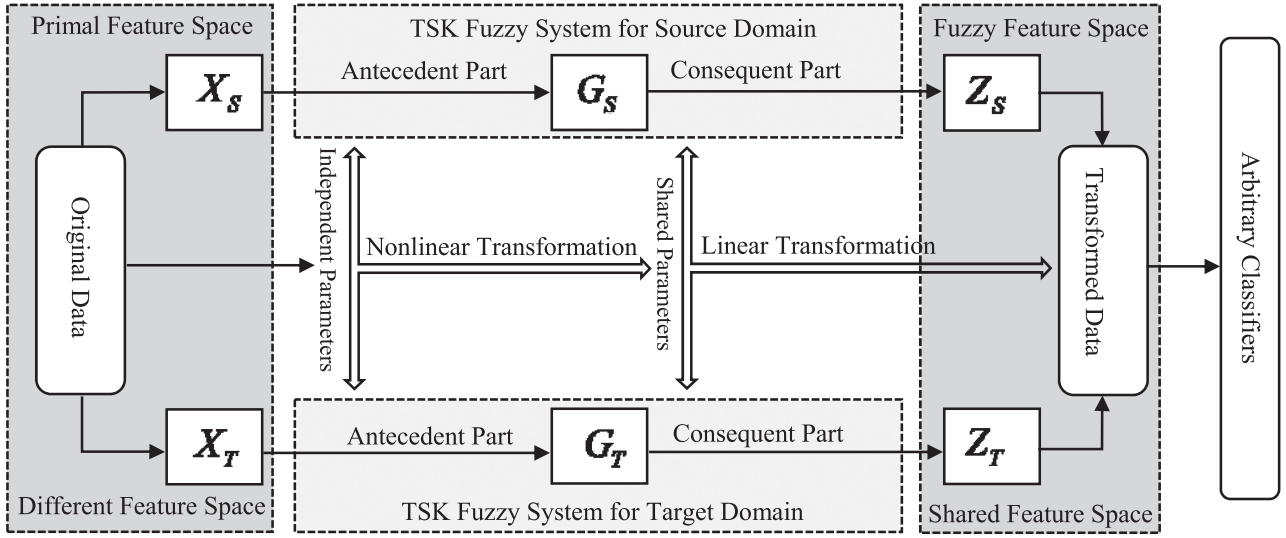


Fig. 1. Flowchart of the proposed method. It consists of three main parts that are highlighted in gray, light gray, and white. The gray part represents the procedure of transformation of the feature space. The light gray part represents the two TSK-FSs. The white part represents the data flow in the whole process. Through the transformation, the generated data can be fed into any classifiers.

the consequent parameter for the m -dimensional output TSK-FS. Then for an example \mathbf{x}_{s_i} or \mathbf{x}_{t_i} in the source domain or the target domain, the transformed data can be represented as follows:

$$\phi(\mathbf{x}_{s_i}) = \mathbf{P}^T \mathbf{g}_{s_i} \quad (12a)$$

$$\phi(\mathbf{x}_{t_i}) = \mathbf{P}^T \mathbf{g}_{t_i}. \quad (12b)$$

By the transformation of the m -dimensional output TSK-FS, all of the data of the two domains in the fuzzy feature space can be represented in matrix form by (13a) and (13b), respectively

$$\mathbf{Z}_S = \mathbf{P}^T \mathbf{G}_S \quad (13a)$$

$$\mathbf{Z}_T = \mathbf{P}^T \mathbf{G}_T. \quad (13b)$$

When the multioutput TSK-FS is regarded as the feature mapping function, two TSK-FSs are applied on the data of the two domains, respectively, as illustrated in Fig. 1. In this article, we assume that the consequent parameters of the two TSK-FSs are shared, whereas the antecedent parameters are different. Take the antecedent parameters of the TSK-FS for the source domain as $\mathbf{C}_S \in R^{K \times d}$ and $\mathbf{D}_S \in R^{K \times d}$ and that for the target domain as $\mathbf{C}_T \in R^{K \times d}$ and $\mathbf{D}_T \in R^{K \times d}$. The \mathbf{C}_S and \mathbf{C}_T represent the center of the membership function in (2). The \mathbf{D}_S and \mathbf{D}_T represent the kernel widths of the membership function in (2). The consequent parameters are solved in Section III-E, whereas the antecedent parameters for both domains are calculated in the following section with unsupervised clustering.

2) *Calculation of Antecedent Parameters:* FCM is a common method to obtain the antecedent parameters of TSK-FS, but its stability is poor due to random initialization in FCM. Hence, the resulting FCM-based TSK-FS is sensitive to the parameters, and the practicability is reduced. In the proposed method TRL-TSK-FS, a deterministic clustering algorithm Var-Part [41] is adopted to obtain the antecedent parameters. First, the Var-Part algorithm is used to cluster data from the source domain and

the target domain, respectively, and two center matrices \mathbf{C}_S and \mathbf{C}_T can be obtained. Note that K is both the number of clusters and the number of rules of TSK-FS.

To cluster the data using the Var-Part algorithm, for a selected cluster C_j , the algorithm computes the variance in each dimension and finds the dimension with the largest variance, such as d_p . Then, let x_{ip} denote the value of example \mathbf{x}_i in feature d_p , and μ_{jp} denote the mean of C_j in feature d_p . Divide C_j into two subclusters C_{j1} and C_{j2} according to the following rule: if x_{ip} is less than or equal to μ_{jp} , assign \mathbf{x}_i to C_{j1} ; otherwise, assign \mathbf{x}_i to C_{j2} . When the abovementioned process is completed on one partition, that is, two clusters are created from a cluster, and the next cluster is then chosen for partitioning by selecting the cluster with the largest within-cluster sum-squared-error. The above is repeated until K clusters are produced.

Once \mathbf{C}_S and \mathbf{C}_T are known, the kernel width matrices \mathbf{D}_S and \mathbf{D}_T are calculated by (14) in the same way as FCM in [38]. The kernel width for each dimension is then scaled to the range $[1, 10]$, which is determined based on extensive experiments in this article

$$(\mathbf{D}_S)_p^k = \sum_{i=1}^{n_s} (x_{s_{ip}} - (\mathbf{C}_S)_p^k)^2 \quad (14a)$$

$$(\mathbf{D}_T)_p^k = \sum_{i=1}^{n_t} (x_{t_{ip}} - (\mathbf{C}_T)_p^k)^2. \quad (14b)$$

In (14), $k = 1, 2, \dots, K$, with K being the number of fuzzy rules, and d is the dimension of the examples with $p = 1, 2, \dots, d$. Equations (14a) and (14b) are used for calculating the kernel widths in the two domains, where $x_{s_{ip}}$ and $x_{t_{ip}}$ represent the values of the examples in the source and target domains for feature d_p , respectively.

C. Distribution Matching

The most commonly used MMD is adopted in this article to match the distributions for the source and target domains in the fuzzy feature space. Based on (9a), (12a), and (12b), the empirical MMD of the marginal distributions between the two domains in the fuzzy feature space can be expressed as follows:

$$\begin{aligned} \text{MMD}_P^2(\mathbf{Z}_S, \mathbf{Z}_T) &= \left\| \frac{1}{n_s} \sum_{i=1}^{n_s} \mathbf{P}^T \mathbf{g}_{s_i} - \frac{1}{n_t} \sum_{j=1}^{n_t} \mathbf{P}^T \mathbf{g}_{t_j} \right\|_{\mathcal{H}}^2 \\ &= \text{Tr}(\mathbf{P}^T \mathbf{G}_X \mathbf{M} \mathbf{G}_X^T \mathbf{P}) \end{aligned} \quad (15a)$$

where $\mathbf{G}_X = [\mathbf{G}_S, \mathbf{G}_T] \in R^{K(d+1) \times (n_s+n_t)}$ and $\mathbf{M} \in R^{(n_s+n_t) \times (n_s+n_t)}$. The specific form of \mathbf{M} is given as follows:

$$\mathbf{M}_{ij} = \begin{cases} \frac{1}{n_s}, & i, j \leq n_s \\ \frac{1}{n_t}, & i, j > n_s \\ -\frac{1}{n_s n_t}, & \text{otherwise.} \end{cases} \quad (15b)$$

Minimizing the distribution distance of the marginal distribution does not guarantee that the distribution distance in the conditional distributions is also minimized. To match the joint distributions between the two domains, conditional distribution matching should also be considered, and the corresponding empirical MMD can be expressed by (16a) based on (9b), (12a), and (12b)

$$\begin{aligned} \text{MMD}_P^2(\mathbf{Z}_S, \mathbf{Z}_T) &= \sum_{c=1}^C \left\| \frac{1}{n_s^{(c)}} \sum_{y_{s_i}=c} \mathbf{P}^T \mathbf{g}_{s_i} - \frac{1}{n_t^{(c)}} \sum_{\hat{y}_{t_i}=c} \mathbf{P}^T \mathbf{g}_{t_i} \right\|_{\mathcal{H}}^2 \\ &= \text{Tr} \left(\mathbf{P}^T \mathbf{G}_X \sum_{c=1}^C \mathbf{M}_c \mathbf{G}_X^T \mathbf{P} \right) \end{aligned} \quad (16a)$$

where $\mathbf{M}_c \in R^{(n_s+n_t) \times (n_s+n_t)}$, $c = 1, 2, \dots, C$, and C is the number of classes. The specific form of \mathbf{M}_c is given in (16b). The pseudo-labels $\hat{\mathbf{Y}}_T = \{\hat{y}_{t_j}\}_{j=1}^{n_t}$ can be predicted by any classifier. The 1 nearest neighbor (1NN) classifier is adopted in this article

$$(\mathbf{M}_c)_{ij} = \begin{cases} \frac{1}{n_s^{(c)} n_s^{(c)}}, & i, j \leq n_s \text{ and } y_{s_i}, y_{s_j} = c \\ \frac{1}{n_t^{(c)} n_t^{(c)}}, & i, j > n_s \text{ and } \hat{y}_{t_{(i-n_s)}}, \hat{y}_{t_{(j-n_s)}} = c \\ -\frac{1}{n_s^{(c)} n_t^{(c)}}, & \begin{cases} i \leq n_s, j > n_s \text{ and } \hat{y}_{s_i}, \hat{y}_{t_{(j-n_s)}} = c \\ i > n_s, j \leq n_s \text{ and } \hat{y}_{t_{(i-n_s)}}, \hat{y}_{s_j} = c \end{cases} \\ 0, & \text{otherwise.} \end{cases} \quad (16b)$$

Equation (15a) and (16a) correspond to the first term of (10). Although (10) minimizes the objective function with respect to feature mapping ϕ , ϕ can be decomposed into nonlinear transformation and linear transformation. The nonlinear transformation can be achieved by the antecedent part of TSK-FS with unsupervised clustering. Therefore, the optimizable parameters are only the linear mapping \mathbf{P} , that is, the consequent parameters of TSK-FS. The minimization of the distribution difference in

the fuzzy feature space can be re-expressed as follows:

$$\min_{\mathbf{P}} \text{Tr}(\mathbf{P}^T \mathbf{G}_X \mathbf{M} \mathbf{G}_X^T \mathbf{P}) + \text{Tr} \left(\mathbf{P}^T \mathbf{G}_X \sum_{c=1}^C \mathbf{M}_c \mathbf{G}_X^T \mathbf{P} \right). \quad (17)$$

D. Discriminant Information and Geometric Property Preservation

In addition to distribution matching, it is also crucial to preserve the discriminant information and geometric properties, which can be impaired during the procedure of distribution matching. If the data transformed by the antecedent part of the TSK-FS is viewed as the intermediate representations in the high-dimensional space, the consequent part of the TSK-FS can be viewed as linear dimensionality reduction on the intermediate representations. Then, the discriminant information and geometric properties can be preserved during the optimization process of the consequent parameters of the TSK-FS.

Since the data in the target domain have no labels, the geometric properties can be preserved by maximizing the variance for the data in the fuzzy feature space, such as the process in PCA. The optimization objective is formulated as follows:

$$\max_{\mathbf{P}} \text{Tr}(\mathbf{P}^T \mathbf{G}_T \mathbf{H}_T \mathbf{G}_T^T \mathbf{P}) \quad (18)$$

where \mathbf{H}_T is the centering matrix, which centralizes the examples so that the covariance matrix can be directly computed as $\mathbf{G}_T \mathbf{H}_T \mathbf{G}_T^T$. $\mathbf{H}_T = \mathbf{I}_{n_t} - (1/n_t) \mathbf{1}_{n_t} \mathbf{1}_{n_t}^T$, where \mathbf{I}_{n_t} is the identity matrix, and $\mathbf{1}_{n_t}$ is the column vector with all ones. Here, n_t is the number of examples in the target domain and the dimension of \mathbf{H}_T .

For the data in the source domain, there are labels for the examples. Discriminant information should be preserved and the optimization objective can be formalized like LDA, that is, maximizing the between-class scatter and minimizing the within-class scatter in the fuzzy feature space. The optimization objective is given as follows:

$$\max_{\mathbf{P}} \frac{\text{Tr}(\mathbf{P}^T \mathbf{S}_b \mathbf{P})}{\text{Tr}(\mathbf{P}^T \mathbf{S}_w \mathbf{P})} \quad (19)$$

where \mathbf{S}_b is between-class scatter matrix, and \mathbf{S}_w is within-class scatter matrix; the specific forms of which are given as follows:

$$\mathbf{S}_b = \sum_{c=1}^C n_s^{(c)} (\mathbf{m}_s^{(c)} - \bar{\mathbf{m}}_s) (\mathbf{m}_s^{(c)} - \bar{\mathbf{m}}_s)^T \quad (20)$$

$$\mathbf{S}_w = \sum_{c=1}^C \mathbf{G}_S^{(c)} \mathbf{H}_S^{(c)} (\mathbf{G}_S^{(c)})^T \quad (21)$$

where $\mathbf{m}_s^{(c)}$ and $\bar{\mathbf{m}}_s$ are the mean of data \mathbf{G}_S belonging to the class c and all classes in the source domain, respectively. The definition of $\mathbf{H}_S^{(c)}$ is similar to \mathbf{H}_T in (18), with replacement of n_t by $n_s^{(c)}$, and $n_s^{(c)}$ is the number of examples belonging to the class c .

Considering the discriminant information and the geometric properties simultaneously, minimizing the information loss in

the fuzzy feature space can be expressed by (22), which corresponds to the second term in (10).

$$\min_{\mathbf{P}} \frac{\text{Tr}(\mathbf{P}^T \mathbf{S}_w \mathbf{P})}{\text{Tr}(\mathbf{P}^T \mathbf{G}_T \mathbf{H}_T \mathbf{G}_T^T \mathbf{P}) + \text{Tr}(\mathbf{P}^T \mathbf{S}_b \mathbf{P})} \quad (22)$$

E. Overall Objective Function and Optimization

The problem of transfer representation learning defined in (10) can be tackled by integrating (18) and (22). By using the $\text{Tr}(\mathbf{P}^T \mathbf{P})$ minimization to avoid overfitting and introducing regularization parameters for each term, the overall objective function can be formulated as (23), shown at the bottom of this page, where α , β , and λ are the regularization parameters for the 2-norm regularization term, discriminant information preserving term, and geometric properties preserving term, respectively. The consequent parameters in \mathbf{P} can be obtained by solving (23). Observing that scaling \mathbf{P} does not influence the results of (23), the objective function can thus be reformulated as follows:

$$\min_{\mathbf{P}} \text{Tr} \left(\mathbf{P}^T \left(\mathbf{G}_X \left(\mathbf{M} + \sum_{c=1}^C \mathbf{M}_c \right) \mathbf{G}_X^T + \alpha \mathbf{I} + \beta \mathbf{S}_w \right) \mathbf{P} \right) \\ \text{s.t. } \text{Tr}(\mathbf{P}^T (\lambda \mathbf{G}_T \mathbf{H}_T \mathbf{G}_T^T + \beta \mathbf{S}_b) \mathbf{P}) = 1 \quad (24)$$

where $\mathbf{I} \in R^{K(d+1) \times K(d+1)}$ is the identity matrix. Equation (24) can be optimized with the Lagrange function as follows:

$$L = \text{Tr} \left(\mathbf{P}^T \left(\mathbf{G}_X \left(\mathbf{M} + \sum_{c=1}^C \mathbf{M}_c \right) \mathbf{G}_X^T + \alpha \mathbf{I} + \beta \mathbf{S}_w \right) \mathbf{P} \right) \\ + \text{Tr}((\mathbf{P}^T (\lambda \mathbf{G}_T \mathbf{H}_T \mathbf{G}_T^T + \beta \mathbf{S}_b) \mathbf{P} - \mathbf{I}) \Phi) \quad (25)$$

where $\Phi = \text{diag}(\varphi_1, \varphi_2, \dots, \varphi_m) \in R^{m \times m}$ is Lagrangian multiplier. By setting $\frac{\partial L}{\partial \mathbf{P}} = 0$, the following equation is obtained:

$$\left(\mathbf{G}_X \left(\mathbf{M} + \sum_{c=1}^C \mathbf{M}_c \right) \mathbf{G}_X^T + \alpha \mathbf{I} + \beta \mathbf{S}_w \right) \mathbf{P} \\ = (\lambda \mathbf{G}_T \mathbf{H}_T \mathbf{G}_T^T + \beta \mathbf{S}_b) \mathbf{P} \Phi. \quad (26)$$

Based on (26), the optimization problem in (23) is transformed into the problem of generalized eigenvalue decomposition. Finding the optimal \mathbf{P} is then reduced to solving (26) for the m smallest eigenvectors. That is, $\varphi_1, \varphi_2, \dots, \varphi_m$ are the m smallest eigenvalues, and $\mathbf{P} = [\mathbf{p}_g^1, \mathbf{p}_g^2, \dots, \mathbf{p}_g^m]$ are the corresponding eigenvectors. Once the consequent parameters \mathbf{P} are obtained, the new data \mathbf{Z}_S and \mathbf{Z}_T generated in the fuzzy feature space can be obtained easily.

TABLE I
DESCRIPTION OF TRL-TSK-FS ALGORITHM

Algorithm: TRL-TSK-FS

Input: Data in the source domain $\mathbf{X}_S = \{\mathbf{x}_{s_i}\}_{i=1}^{n_s}$ and the corresponding labels $\mathbf{Y}_S = \{y_{s_i}\}_{i=1}^{n_s}$; the data in the target domain $\mathbf{X}_T = \{\mathbf{x}_{t_i}\}_{i=1}^{n_t}$; trade-off parameters α , β and λ ; the number of fuzzy rules K ; the dimension of the fuzzy feature space m ; the number of iterations for joint distribution adaptation T ; the classifier used to train the new data and label the data in target domain.

Output: New data in the fuzzy feature space \mathbf{Z}_S and \mathbf{Z}_T .

Procedure TRL-TSK-FS

- 1: Calculate the antecedent parameters \mathbf{C}_S , \mathbf{C}_T , \mathbf{D}_S and \mathbf{D}_T for the data \mathbf{X}_S and \mathbf{X}_T based on algorithm Var-Part and (12).
- 2: Calculate \mathbf{G}_S and \mathbf{G}_T using \mathbf{C}_S , \mathbf{C}_T , \mathbf{D}_S and \mathbf{D}_T based on 7(a) – 7(c).
- 3: **For** $t \leftarrow 1, 2, \dots, T$ **do**
- 4: Update \mathbf{M} and \mathbf{M}_c based on (14) and (16).
- 5: Update \mathbf{S}_b and \mathbf{S}_w based on (20) and (21)
- 6: Update \mathbf{P} based on (25) using generalized eigenvalue decomposition.
- 7: Update \mathbf{Z}_S and \mathbf{Z}_T based on (11a) and (11b).
- 8: Train the selected classifier using \mathbf{Z}_S and \mathbf{Z}_T , and update the pseudo labels for the data in the target domain $\hat{\mathbf{Y}}_T = \{\hat{y}_{t_i}\}_{i=1}^{n_t}$.
- 9: **end for**

After getting the new representations through transfer learning, arbitrary classifiers can be applied to classify the data in the target domain. Due to the existence of joint distribution adaptation, the process of optimization is iterative. Whenever new representations are obtained, the classifier is used to update the labels of the data in the target domain, and then the abovementioned process is repeated. The description of the TRL-TSK-FS algorithm is summarized in Table I.

F. Computational Complexity

Based on the abovementioned discussion, we will analyze the efficiency of the proposed method. The computational complexity of the TRL-TSK-FS algorithm presented in Table I is analyzed using the big O notation. Denote the number of examples as N with $N = n_s + n_t$, the dimension of the original data as d , and the number of fuzzy rules as K . For step 1, the computational complexity for calculating \mathbf{C}_S , \mathbf{C}_T , \mathbf{D}_S , and \mathbf{D}_T is $O(2dNK)$. For step 2, the computational complexity for calculating the \mathbf{G}_S and \mathbf{G}_T is $O(2dNK + 2NK)$ based on (2), (3), (4), and (7b). Denote the number of iterations for

$$\min_{\phi} \text{Distance}(\mathbf{X}_S, \mathbf{X}_T | \phi) + \text{Info_loss}(\mathbf{X}_S, \mathbf{X}_T | \phi) \\ = \min_{\mathbf{P}} \frac{\text{Tr}(\mathbf{P}^T \mathbf{G}_X (\mathbf{M} + \sum_{c=1}^C \mathbf{M}_c) \mathbf{G}_X^T \mathbf{P}) + \alpha \text{Tr}(\mathbf{P}^T \mathbf{P}) + \beta \text{Tr}(\mathbf{P}^T \mathbf{S}_w \mathbf{P})}{\lambda \text{Tr}(\mathbf{P}^T \mathbf{G}_T \mathbf{H}_T \mathbf{G}_T^T \mathbf{P}) + \beta \text{Tr}(\mathbf{P}^T \mathbf{S}_b \mathbf{P})} \\ = \min_{\mathbf{P}} \frac{\text{Tr}(\mathbf{P}^T (\mathbf{G}_X (\mathbf{M} + \sum_{c=1}^C \mathbf{M}_c) \mathbf{G}_X^T + \alpha \mathbf{I} + \beta \mathbf{S}_w) \mathbf{P})}{\text{Tr}(\mathbf{P}^T (\lambda \mathbf{G}_T \mathbf{H}_T \mathbf{G}_T^T + \beta \mathbf{S}_b) \mathbf{P})} \quad (23)$$



Fig. 2. Example of the images in the Office-Caltech dataset.

joint distribution adaptation as T . The computational complexity of step 4 for constructing MMD matrices is $O(TCN^2)$. In step 5, the computational complexity for calculating S_b and S_w is $O(TCN + TNK^2d^2 + TN^2Kd)$. Denote the dimension of fuzzy feature space as m , the computational complexity of generalized eigenvalue decomposition in step 6 is $O(Tmk^2d^2)$. The computational complexity of constructing the data Z_S and Z_T in step 7 is $O(TNmd)$. Based on the abovementioned analysis, the major computation cost lies in MMD matrices construction in step 4, matrix multiplication in step 5, and generalized eigenvalue decomposition in step 6. The computational cost can increase exponentially with the size of the examples. With the increase in number of fuzzy rules, the computational cost also increases exponentially, especially when dealing with high-dimensional data. Let $a = \max(T, C, m)$ and $b = \max(N, Kd)$, the maximum overall computational cost is $O((4a^2 + 2ab)b^2)$.

IV. EXPERIMENTS

A. Datasets

The experiments are conducted to evaluate the effectiveness of the proposed algorithm on commonly used image and text datasets for transfer learning. They are the image transfer dataset Office-Caltech and text transfer dataset NG20.

The Office-Caltech dataset is composed of the Office dataset and the Caltech-256 dataset. Some example pictures of Office-Caltech dataset are shown in Fig. 2. Office [42], [43] is a benchmark dataset widely used for transfer learning, which contains 31 different categories. The images in the Office dataset come from three domains: AMAZON (images downloaded from online website, that is, www.amazon.com), Webcam (low-resolution images captured by a simple Web camera), and DSLR (high-resolution images captured by a digital single-lens reflex camera). Caltech-256 [44] is a well-known object recognition dataset, which contains 256 different categories. In the experiments, the settings are the same as those used in [16]. The four domains are AMAZON, Webcam, DSLR, Caltech-256, denoted by A, W, D, and C, respectively, each with ten categories. The speeded-up robust features [45] feature of the selected images is extracted, and the codebook is constructed by K-means to represent the images in the form of 800-bins with

TABLE II
CONSTRUCTION OF NG20

	Domain-A	Domain-B
COMP	graphics os.ms-windows.misc	sys.ibm.pc.hardware sys.mac.hardware
REC	autos motorcycles	sport.baseball sport.hockey
SCI	crypt electronics	med space
TALK	politics.guns politics.mideast	politics.misc religion.misc

bag of visual work model. Finally, all data are standardized. Two arbitrary domains are selected from the four domains as the source domain and the target domain, and 12 distinct transfer tasks are constructed from the four domains, e.g., $C \rightarrow A$, $C \rightarrow W$, ..., $D \rightarrow W$.

NG20 is a text transfer dataset constructed from the 20-Newsgroup, which contains more than 20 000 examples with six major categories and 20 subcategories [46], [47]. In the experiments, four major categories are selected, which are denoted as COMP, REC, SCI, and TAKL. Table II shows the details of the selection of subcategories in the experiment. Two arbitrary major categories are selected to a transfer task, and 12 transfer tasks are constructed with the selected documents. For example, when COMP and REC are selected, the Domain-A of them are combined as the data of the source domain, and the Domain-B of them are combined as the data of the target domain. This transfer task is represented as $COMP \rightarrow REC$. In the same way, if Domain-A is treated as the data in the target domain and Domain-B is treated as the data in the source domain, the transfer task is represented as $REC \rightarrow COMP$. Hence, 12 transfer tasks are constructed from the four major categories. In order to improve the efficiency of the algorithm, the deep learning method Doc2vec [48] is used to extract the features for all selected documents. This method is used to represent every document with different lengths as vectors with fixed length in the form of unsupervised learning. Before using the Doc2vec model to train document vectors, word segmentation, stop words removing, stemming and lemmatization operations are performed to preprocess the documents [49]. Documents vectors with 200 dimensions are extracted finally.

In the experiments, both the labeled data in the source domain and the unlabeled data in the target domain are used to train the transfer model, the transferable representations are then obtained, based on which the data in the source domain are treated as training data and the data in the target domain are treated as the test data. The accuracy on the test data in the target domain is reported.

B. Experimental Settings

1) *Comparison Algorithms*: In the experiments, two non-transfer and five transfer methods are used as the comparison algorithms. The two nontransfer algorithms are 1NN and TSK-FS. 1NN is used as the baseline for all the algorithms under comparison and is denoted as 1NN (raw). 1NN is also

TABLE III
ACCURACY ON IMAGE TRANSFER DATASET OFFICE-CALTECH (%)

Transfer Tasks	1NN (raw)	1NN (PCA)	TSK (raw)	TCA	GFK	JDA	TJM	SCA	TRL-TSK (K=3)
C→A	23.70	36.95	52.40	45.82	41.02	44.78	46.76	43.74	58.46
C→W	25.76	32.54	47.46	30.51	40.68	41.69	38.98	33.56	49.83
C→D	25.48	38.22	44.59	35.67	38.85	45.22	44.59	39.49	46.50
A→C	26.00	34.73	43.99	40.04	40.25	39.36	39.45	38.29	45.06
A→W	29.83	35.59	38.98	35.25	38.98	37.97	42.03	33.90	50.85
A→D	25.48	27.39	43.95	34.39	36.31	39.49	45.22	34.21	46.50
W→C	19.86	26.36	33.84	29.92	30.72	31.17	30.19	30.63	38.56
W→A	22.96	29.35	37.68	28.81	29.75	32.78	29.96	30.48	45.62
W→D	59.24	77.07	82.80	85.99	80.89	89.17	89.17	92.36	94.27
D→C	26.27	29.65	30.81	32.06	30.28	31.52	31.43	32.32	36.24
D→A	28.50	32.05	33.82	31.42	32.05	33.09	32.78	33.72	45.30
D→W	63.39	75.93	81.36	86.44	75.59	89.49	85.42	88.81	94.24
Average	31.37	39.65	47.64	43.03	42.95	46.31	46.33	44.29	54.29

applied on the processed data with PCA for fair comparison, which is denoted as 1NN (PCA). TSK-FS is used to verify the effectiveness of the proposed transfer strategy. With increasing number of rules, ill-posed problem may arise when the least square method is used to solve for the consequent parameters. Hence, the ridge regression technique is adopted in the experiments to alleviate the problem [50].

The five nontransfer algorithms are TCA, geodesic flow kernel (GFK) [43], JDA, TJM, and SCA. These five methods are all transfer representation learning methods, where only the transferable features are learned after the algorithms. 1NN is then used to classify the new representations obtained by these five algorithms.

2) *Hyperparameter Settings of Comparison Algorithms:* Since there are no labels for the data in the target domain, cross validation cannot be used to find the optimal parameters for all the comparison algorithms. Hence, the parameters are optimized by grid search. The number of rules of TSK-FS is optimally set by searching the grid $k = \{3, 5, 10\}$. For the five transfer methods, the dimension of subspace is optimally set by searching the grid $m = \{10, 20, \dots, 100\}$. Except for special declaration, all the tradeoff parameters and regularization parameters of the algorithms are optimally set by searching the grid $\lambda = \{0.01, 0.1, 1, 10, 100\}$. For all the algorithms involving joint distribution adaptation, the number of iterations is set as $T = 5$ because with pseudo-labels, it requires several iterations to improve the model accuracy. For the tradeoff parameter of SCA, which controls the balance of the total scatter and the between-class scatter, it is optimally set with the grid $\beta = \{0.1, 0.3, 0.5, 0.8, 1\}$.

3) *Hyperparameter Settings of TRL-TSK-FS:* For the proposed method TRL-TSK-FS, the tradeoff parameters α , β , and λ and the number of iterations for joint distribution adaptation are optimally set in the same way as those applied to the comparison algorithms. Since deterministic clustering algorithm is adopted to calculate the antecedent part of the TSK-FS, there is no parameter to be optimized, unlike the commonly used method FCM. An important parameter of TRL-TSK-FS is the number of rules that can influence the dimension of the new representations in the

fuzzy feature space and the effect of nonlinear transformation for domain adaptation. This parameter is optimally set with the search grid $K = \{3, 4, 5, \dots, 10\}$. In real applications, even not all labels of the data in the target domain are available, and a part of labeled data is enough for choosing the appropriate parameters.

C. Result Analysis

1) *On Image Dataset:* Table III illustrates the accuracy of all the algorithms on the image transfer dataset Office-Caltech. Except TSK and TRL-TSK-FS, the reported accuracies of 1NN (raw), 1NN (PCA), TCA, and GFK are from [8], and that of the other algorithms are from their corresponding original papers. For the algorithms involving kernel functions, the reported results have considered linear kernel function and radial basis kernel function. The method TSK (raw) follows the training strategy of traditional machine learning methods, which treat the data in the source domain as the training data and the data in the target domain as the test data. TRL-TSK ($K = 3$) denotes the accuracy of the proposed method TRL-TSK-FS when the number of rule is 3. The optimal results for each transfer task in Table III are highlighted in bold. It can be observed that the accuracy of TRL-TSK-FS is superior to all the other algorithms on Office-Caltech dataset. The accuracy of TJM is the highest in all comparing algorithms considering the average accuracy of 12 transfer tasks. The average accuracy of the proposed method is nearly higher 8% than that of TJM under of rule number of 3. The average accuracy of the proposed method is higher about 23% than that of 1NN (raw).

2) *On Text Dataset With Linear Kernel Function:* Table IV illustrates the accuracy of all algorithms on text transfer dataset NG20. Since the NG20 dataset is constructed in this article, the results of all algorithms are obtained according to the experimental settings presented in Section IV-B. Linear kernel function is selected for the algorithms involving kernel functions. The optimal results for each transfer task in Table IV are highlighted in bold. The training strategy of methods 1NN (raw), 1NN (PCA), and TSK (raw) are the same as TSK (raw) on the

TABLE IV
ACCURACY ON TEXT TRANSFER DATASET NG20 WITH LINEAR KERNEL FUNCTION (%)

Transfer Tasks	1NN (raw)	1NN (PCA)	TSK (raw)	TCA	GFK	JDA	TJM	SCA	TRL-TSK (K=3)
COM→REC	57.21	69.32	86.06	81.74	86.87	92.76	90.45	90.43	97.44
REC→COM	59.99	61.00	68.43	82.66	73.96	93.61	90.92	93.81	94.04
COM→SCI	56.73	62.21	66.37	59.99	73.33	81.44	78.66	76.03	81.92
SCI→COM	53.72	57.44	62.75	64.23	67.71	75.49	73.66	75.64	75.92
COM→TALK	81.81	92.29	95.61	92.95	95.55	96.48	95.43	95.67	96.80
TALK→COM	86.27	92.52	95.38	94.98	95.06	96.43	96.24	95.43	95.90
REC→SCI	51.51	58.41	65.62	65.14	72.80	87.46	81.96	89.17	88.36
SCI→REC	58.50	58.95	63.47	65.82	67.91	85.84	85.43	80.79	87.38
REC→TALK	55.45	68.88	80.89	85.51	88.31	90.31	85.54	82.95	91.55
TALK→REC	71.30	72.13	81.31	90.62	83.73	90.80	91.32	89.36	92.81
SCI→TALK	62.51	76.15	80.77	78.52	79.79	78.88	79.02	76.24	84.64
TALK→SCI	69.93	72.42	80.34	84.08	80.13	82.80	84.86	85.86	86.88
Average	63.74	70.14	77.25	78.85	80.43	87.69	86.12	85.95	89.47

TABLE V
ACCURACY ON TEXT TRANSFER DATASET NG20 WITH RADIAL BASIS FUNCTION (%)

Transfer Tasks	TCA	JDA	TJM	SCA	TRL-TSK
COM→REC	81.18	93.17	87.66	87.46	97.44
REC→COM	81.57	93.28	90.44	84.56	94.04
COM→SCI	59.15	79.70	76.16	62.52	81.92
SCI→COM	63.51	75.62	74.09	67.00	75.92
COM→TALK	92.35	96.21	95.22	94.21	96.80
TALK→COM	95.40	96.43	96.09	94.20	95.90
REC→SCI	64.86	87.15	79.19	73.35	88.36
SCI→REC	64.91	86.57	85.38	56.43	87.38
REC→TALK	84.57	89.84	83.19	78.21	91.55
TALK→REC	90.22	92.18	90.67	88.24	92.81
SCI→TALK	78.58	80.56	77.63	78.70	84.64
TALK→SCI	83.74	82.77	85.18	79.08	86.88
Average	78.34	87.79	85.08	78.66	89.47

image dataset. The method 1NN (raw) classifies the original data directly, whereas the method 1NN (PCA) classifies the data after dimensionality reduction using PCA. It is obvious that the average accuracy of TRL-TSK-FS on 12 transfer tasks outperforms all the other algorithms on NG20 dataset. The average accuracy of the proposed method is nearly higher 2% than that of JDA, which achieved best results in all comparing algorithms. The first two algorithms 1NN (raw) and 1NN (PCA) have the lowest average accuracy in all algorithms. However, the accuracy of 1NN classifier increased by about 7% after PCA dimensionality reduction. The average accuracy of TSK (raw) are higher than that of 1NN (raw). It is because that the learning ability of TSK-FS exceeds that of 1NN. It is unreasonable to make direct comparison between TSK-FS and the other algorithms. Even so, the performance of TRL-TSK-FS with 1NN classifier is still superior to TSK-FS classifier, which verifies the effectiveness of transfer representation learning.

3) *On Text Dataset With Radial Basis Function:* Table V shows the results of the algorithms involving kernel functions, and the kernel function is set as radial basis function (RBF). It is still an open issue to select appropriate kernel functions for these algorithms. Besides linear kernel function, the RBF

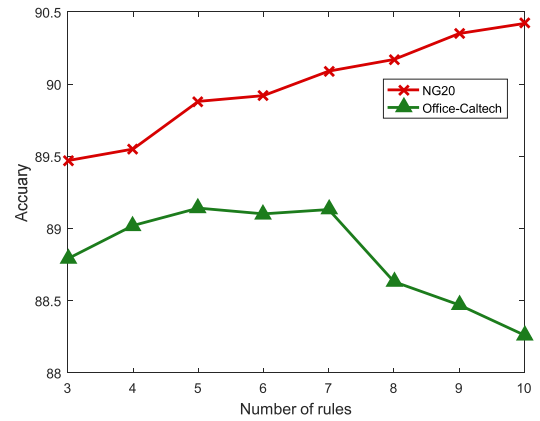


Fig. 3. Variation in accuracy with the number of rules.

is the commonly used nonlinear kernel function for feature transformation. Similar to that in [8], [17], and [40], the form of RBF is $\exp(-\|\mathbf{a} - \mathbf{b}\|_2^2 / \sigma^2)$, where the kernel width σ is set to the median distance between samples in the aggregate domain in the experiments

$$\sigma = \text{median}(\|\mathbf{a} - \mathbf{b}\|_2^2) \quad \forall \mathbf{a}, \mathbf{b} \in \mathbf{X}_S \cup \mathbf{X}_T. \quad (27)$$

Similar to the kernel functions, the antecedent part in of TRL-TSK-FS also plays the role of nonlinear transformation. There are no parameters need to be optimized in the antecedent part of TRL-TSK-FS. Unlike kernel functions whose selection is blind and difficult, the fuzzy mapping of the antecedent part is more intuitive and interpretable. When the other experimental settings are the same as that in the previous experiments, the results of TCA, JDA, TJM, and SCA are shown in Table V. It can be seen that the proposed method clearly demonstrates superiority. The performance of SCA is poor with the RBF kernel function on the text dataset, which may be due to the reason that the selected kernel width is not appropriate for the text data. The abovementioned discussion verifies the effectiveness of TSK-FS for transfer representation learning.

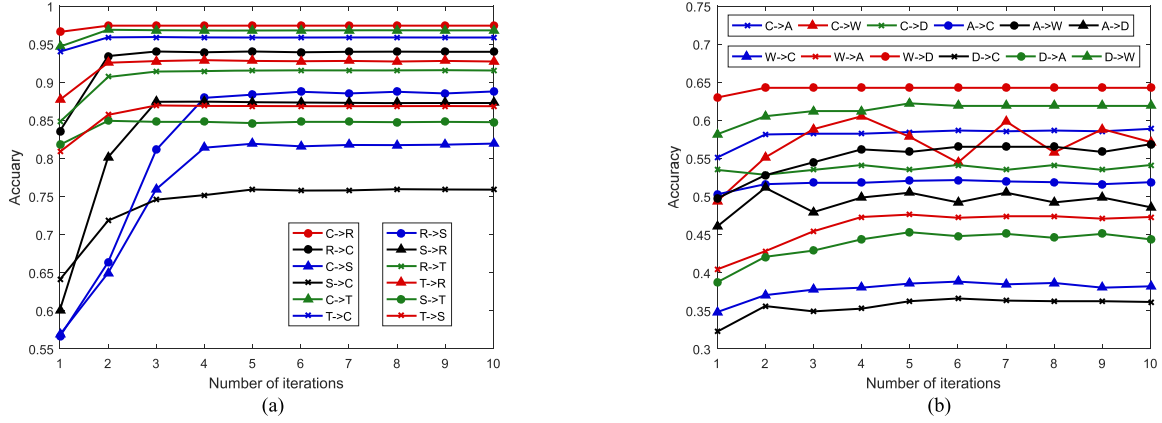


Fig. 4. Convergence analysis on the image and text dataset.

D. Parameters Analysis

1) *Number of Fuzzy Rules*: According to the analysis presented in Section III-F, computational complexity can increase exponentially with the number of rules of TSK-FS. The effect is studied by running the algorithms with different number of rules. Fig. 3 shows the average accuracy on the image and text datasets. The maximum number of rules is set as ten in the experiments. For the image dataset Office-Caltech, high accuracy can be achieved with five, six, and seven rules. The accuracy decreases when the number of rules exceeds seven. For the image dataset NG20, the accuracy keeps increasing with the number of rules. Since the image dataset is of higher dimension than the text dataset, 800 versus 200, the number of parameters and the complexity of the model for the image dataset increases more rapidly, which may lead to overfitting when the number of rules exceeds seven. It can be concluded from Fig. 3 that for high-dimensional dataset, a few rules are enough; whereas for low-dimensional dataset, the performance increases steadily with increasing number of rules.

Indeed, the choice of the number of fuzzy rules has been a topic of study [51]–[53]. Most existing techniques can be used to determine an appropriate value for the number of fuzzy rules and can be used directly in the proposed method. In addition, the generation of antecedents heavily depends on the clustering algorithms adopted, and it is feasible to choose the algorithms that can automatically determine the number of clusters, e.g., ISODATA and infinite Gaussian mixture model. But these methods are usually heuristic and may introduce more hyperparameters into the methods. In this article, the main purpose is to develop a more interpretable feature representation learning method for transfer learning task. A concise fuzzy rule base is thus preferred to implement this task since the less the number of fuzzy rules, the better the interpretability of the model is. In our experiments, three rules are adopted for all datasets to maintain a better interpretability, which is found to be enough to achieve promising performance. Of course, some existing techniques can also be used to determine the appropriate number of rules and further improve the classification accuracy of the proposed method, but if a large number of rules are involved, the

interpretability will be weakened. Hence, the number of rules is set to three for the proposed method in the experiments.

2) *Number of Iterations for Joint Distribution Adaptation*: Fig. 4(a) and 4(b) shows the accuracy under different numbers of iterations for each task on the image and text datasets, respectively. It can be seen from Fig. 4(a) that the proposed method exhibits good convergence on the 12 tasks for the image dataset. The accuracy is fairly stable when the number of iterations exceeds five. Fig. 4(b) shows that for the text dataset, good convergence is exhibited for most tasks after five iterations, only the accuracy of the task $C \rightarrow W$ fluctuates over the iterations, whereas the amplitude of fluctuation decreases as the number of iterations increases. Therefore, it is reasonable to set the number of iterations to five in the experiments.

3) *Dimensionality of Fuzzy Feature Space*: Given the m -dimensional output TSK-FS, the dimensionality of the fuzzy feature space is set as m . Fig. 5(a) and (b) shows the accuracy with different dimensionalities of the fuzzy feature space for each task on the image and text dataset. In the figures, four representative tasks are only selected for both datasets in order to visualize the trends and variations more clearly. Although the four tasks in Fig. 5(a) show different trends of classification accuracy change as the dimensionality increases for different transfer tasks, they are relatively stable and the optimal accuracy can be obtained within 100 dimensions. Fig. 5(b) shows that for the text dataset, the classification accuracy of four tasks shows downward trends with the increase in dimensionality. It can be seen that on the text dataset, the low-dimensional feature space can get better classification accuracy.

4) *Tradeoff Parameters*: The tradeoff parameters α , β , and λ are analyzed based on the experimental results. Take six transfer tasks on the image dataset as the standard, Fig. 6(a)–(f) shows the effect of parameters on classification accuracy under different tasks. Fig. 6(a) and (b) shows the effect of parameter α on the classification accuracy with the other parameters fixed. It can be seen that $\alpha = 0.1$ can get promising results, and it can be obtained in most cases. Similarly, Fig. 6(c) and (d) indicates that $\beta = 0.01$ is the optimal value, and Fig. 6(e) and (f) indicates that with $\lambda = 0.01$ or $\lambda = 0.1$, better classification results can be achieved.

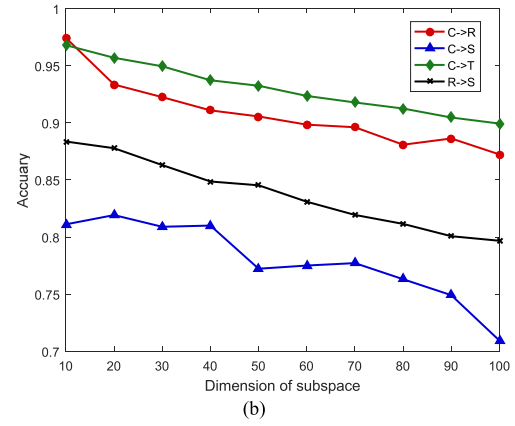
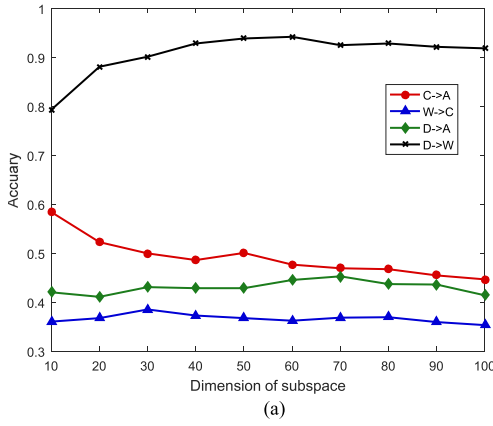


Fig. 5. Analysis of the effect of dimensionality.

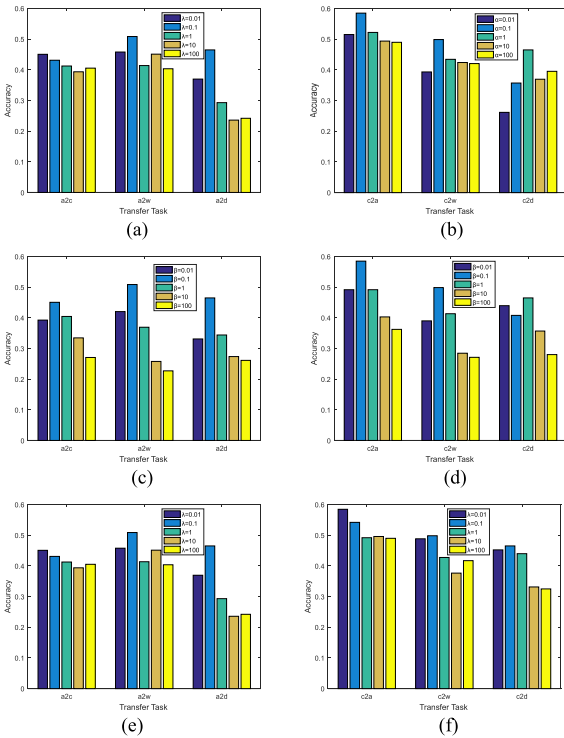


Fig. 6. Analysis of tradeoff parameters.

E. Comparison With Fuzzy Transfer Models

The comparison algorithms in Tables III and IV are all classical feature-based domain adaptation methods. There are also many fuzzy-technique-based transfer learning algorithms proposed [21]–[29], [54], which have been described and reviewed in Section I. Except for the method presented in [54], all these methods belong to model-based transfer learning, which is different from the feature-based transfer learning paradigm presented in this article. In the proposed method, the invariant representations can be learned across domains, and the selection of final classifiers is arbitrary. Hence, making a fair comparison between the two schools of algorithms is not feasible.

In [54], Liu *et al.* proposed an unsupervised heterogeneous domain adaptation method based on fuzzy equivalence relations

TABLE VI
ACCURACY OF METHODS F-HeUDA AND TRL-TSK-FS (%)

Transfer Tasks	Source Domain	Target Domain	F-HeUDA	TRL-TSK
CancerD→CancerO	CancerD	CancerO	86.09	85.07
CancerO→CancerD	CancerO	CancerD	87.35	90.86
Aus→German	Aus	German	51.90	68.80
German→Aus	German	Aus	56.23	70.43

(F-HeUDA), where the transfer paradigm is almost the same as that in the proposed method. The main difference is that F-HeUDA is based on fuzzy equivalence relations and our TRL-TSK-FS is based on fuzzy feature space and MMD. Since the computational complexity of F-HeUDA is very high for high-dimensional datasets, four low-dimensional datasets are selected to perform the comparison. The selected datasets are the same as that used in [54], and the constructed transfer tasks are almost identical. The only difference is that all the data in the datasets are used in our experiments, whereas only a part of the data are used in [54]. A detailed description and the construction of the datasets and the transfer tasks can be found in [54]. Since the proposed TRL-TSK-FS can only deal with heterogeneous domain adaptation, PCA is applied to the datasets for TRL-TSK-FS in the experiments. Table VI shows the performance of domain adaptation of these two methods. It is obvious that TRL-TSK-FS is superior to F-HeUDA and that F-HeUDA outperforms TRL-TSK-FS for the transfer task CancerD→CancerO.

F. Interpretability Analysis

The interpretability of the fuzzy system is mainly attributed to rule-based mechanism and human-like fuzzy inference. TSK-FS can be regarded as a regression model or a classifier. In the proposed method, it is regarded as a feature transformation method to construct the fuzzy feature space. Compared with kernel methods, TSK-FS make the process of feature transformation more interpretable, where the transformation process can be interpreted as a set of rules.

In the proposed method, multioutput TSK-FS is adopted, where the multiple outputs represent the new features. The rules

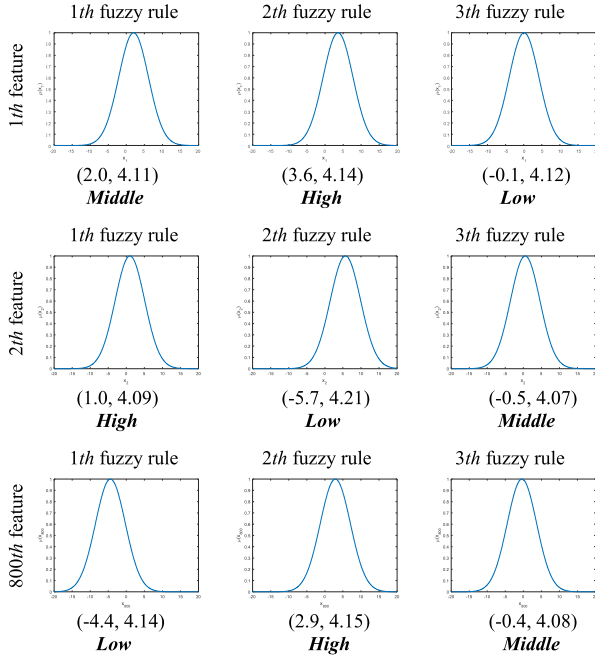


Fig. 7. Membership functions and the possible linguistic explanation of each fuzzy set for the first (top line), second (middle line), and the last dimension (bottom line) for the C→A task.

for feature transformation can be formulated as follows:

$$\begin{aligned}
 \text{IF: } x_1 \text{ is } A_1^k(c_1^k, \delta_1^k) \wedge x_2 \text{ is } A_2^k(c_2^k, \delta_2^k) \wedge \dots \\
 \wedge x_d \text{ is } A_d^k(c_d^k, \delta_d^k), \\
 \text{THEN: } f^k(\mathbf{x}) = [(p_0^k)^1 + (p_1^k)^1 x_1 + (p_2^k)^1 x_1 + \dots \\
 + (p_d^k)^1 x_1 \\
 (p_0^k)^2 + (p_1^k)^2 x_1 + (p_2^k)^2 x_1 + \dots + (p_d^k)^2 x, \dots, \\
 (p_0^k)^m + (p_1^k)^m x_1 + (p_2^k)^m x_1 + \dots + (p_d^k)^m x]. \quad (28)
 \end{aligned}$$

The advantage of fuzzy systems is that they can be explained as natural semantics. In the TSK-FS, each fuzzy set $A_i^k(c_i^k, \delta_i^k)$ for the rule k and dimension i can be interpreted with a linguistic term. Since the number of fuzzy rules in our experiments is three, there are three clustering centers for each dimension. Then, the linguistic descriptions can be *Low*, *Middle*, and *High* corresponding to the fuzzy sets with the smallest, mid-sized, and the largest centers, respectively. The given linguistic description is only a possible explanation for the If-Part of a fuzzy rule. Other descriptions can be applied depending on the application scenarios.

The interpretability of the TSK-FS for the source domain can be illustrated by taking the transfer task C→A as an example. Since the data in this transfer task have 800 dimensions, we only illustrate 3 dimensions among them, i.e., the first, second, and the last dimensions (i.e., 800th). With the Gaussian membership function adopted in the proposed method, the corresponding fuzzy sets are shown in Fig. 7. Taking the first row of Fig. 7 as an example, the center and variance of the first dimension in the first fuzzy rule are 2.0 and 4.11, respectively, where the center value “2.0” ranks the second among the three centers, i.e.,

TABLE VII
RULE BASE GENERATED FOR THE C→A TRANSFER TASK

The Rule Base of the TSK-FS for the source domain	
Rule 1:	
IF:	the 1th feature is <i>Middle</i> , and the 2th feature is <i>High</i> , and, and the 800th feature is <i>Low</i> .
Then:	the 1th output is $0.0065+0.0134x_1+0.0042x_2+\dots-0.0102x_{800}$, and the 2th output is $0.0134+0.0004x_1-0.0337x_2+\dots+0.0095x_{800}$, and, and the 10th output is $-0.0243+0.0159x_1+0.0001x_2+\dots-0.0141x_{800}$.
Rule 2:	
IF:	the 1th feature is <i>High</i> , and the 2th feature is <i>Low</i> , and, and the 800th feature is <i>High</i> .
Then:	the 1th output is $0.0068+0.0124x_1+0.0042x_2+\dots+0.0250x_{800}$, and the 2th output is $0.0222-0.0009x_1-0.0337x_2+\dots+0.0230x_{800}$, and, and the 10th output is $-0.0240+0.0160x_1+0.0001x_2+\dots-0.0041x_{800}$.
Rule 3:	
IF:	the 1th feature is <i>Low</i> , and the 2th feature is <i>Middle</i> , and, and the 800th feature is <i>Middle</i> .
Then:	the 1th output is $0.0073+0.0130x_1+0.0038x_2+\dots-0.0149x_{800}$, and the 2th output is $0.0228-0.0001x_1-0.0323x_2+\dots-0.0325x_{800}$, and, and the 10th output is $-0.0239+0.0158x_1-0.0005x_2+\dots+0.0182x_{800}$.

2.0, 3.6, and -0.1 , and hence this fuzzy set is assigned with the term *Middle*. Once all the fuzzy sets are assigned with linguistic terms, then the fuzzy system can be interpreted with fuzzy rules. The descriptions of fuzzy rule base are illustrated in Table VII, and the feature transformation is described as a set of rules.

V. CONCLUSION

A novel transfer representation learning method TRL-TSK-FS based on TSK-FS is proposed in this article. From the aspect of TSK-FS, unlike the other fuzzy transfer learning methods where TSK-FS is treated as a classifier, TSK-FS is regarded as a feature learning method in the proposed method. From the aspect of feature learning, unlike the traditional methods that utilize kernel functions, TRL-TSK-FS performs nonlinear transformation by fuzzy mapping. There are various methods to construct the fuzzy mapping, and a deterministic clustering method is adopted in this article to avoid the problem of initialization sensitivity. The results of the experiments on image and text datasets show that the proposed method is superior to a number of state-of-art transfer representation learning methods.

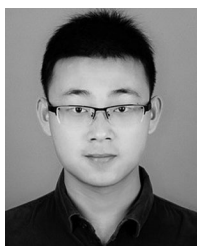
Future research will be conducted to investigate transfer representation learning based on TSK-FS from the following aspects. In the proposed TRL-TSK-FS, the consequent parts of the TSK-FS of the source and the target domains are shared, which limits the application of TRL-TSK-FS to homogeneous transfer learning only [55]. Heterogeneous transfer learning will be explored in the future to enhance the transfer learning ability of the TSK-FS. Furthermore, PCA and LDA are used to preserve the data geometric properties in the proposed method, which can only preserve the global structures of the data. Other techniques that can preserve the local structures of the data, such

as locality preserving projections [56], will thus be investigated. In addition, how to improve the scalability of the proposed method is also a significant future work.

REFERENCES

- [1] C. B. Do, "Transfer learning for text classification," in *Proc. Adv. Neural Inf. Process. Syst.*, 2006, vol. 18, pp. 299–306.
- [2] J. Tahmoresnezhad and S. Hashemi, "Visual domain adaptation via transfer feature learning," *Knowl. Inf. Syst.*, vol. 50, pp. 1–21, 2016.
- [3] D. Ravi *et al.*, "Deep learning for health informatics," *IEEE J. Biomed. Health Inform.*, vol. 21, no. 1, pp. 4–21, Jan. 2017.
- [4] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
- [5] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, vol. 312, pp. 135–153, 2018.
- [6] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, no. 10, pp. 1345–1359, Oct. 2010.
- [7] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," in *Proc. 32nd Int. Conf. Int. Conf. Mach. Learn.*, 2015, pp. 97–105.
- [8] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer joint matching for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 1410–1417.
- [9] L. Bruzzone and M. Marconcini, "Domain adaptation problems: a DASVM classification technique and a circular validation strategy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 770–787, May 2010.
- [10] L. Duan, I. W. Tsang, D. Xu, and T. S. Chua, "Domain adaptation from multiple sources via auxiliary classifiers," in *Proc. Int. Conf. Mach. Learn.*, Montreal, QC, Canada, Jun. 2009, pp. 289–296.
- [11] J. Yang, R. Yan, and A. G. Hauptmann, "Cross-domain video concept detection using adaptive SVMS," in *Proc. 15th ACM Int. Conf. Multimedia*, 2007, pp. 188–197.
- [12] L. Duan, I. W. Tsang, D. Xu, and S. J. Maybank, "Domain transfer SVM for video concept detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 1375–1381.
- [13] B. Quanz and J. Huan, "Large margin transductive transfer learning," in *Proc. ACM Conf. Inf. Knowl. Manage.*, 2009, pp. 1327–1336.
- [14] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE Trans. Neural Netw.*, vol. 22, no. 2, pp. 199–210, Feb. 2011.
- [15] M. Long, J. Wang, G. Ding, S. J. Pan, and P. S. Yu, "Adaptation regularization: A general framework for transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 5, pp. 1076–1089, May 2014.
- [16] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer feature learning with joint distribution adaptation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Sydney, NSW, Australia, 2013, pp. 2200–2207.
- [17] M. Ghifary, D. Balduzzi, W. B. Kleijn, and M. Zhang, "Scatter component analysis: A unified framework for domain adaptation and domain generalization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, pp. 1414–1430, 2017.
- [18] Y. Zhang, H. Ishibuchi, and S. Wang, "Deep Takagi–Sugeno–Kang fuzzy classifier with shared linguistic fuzzy rules," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 3, pp. 1535–1549, Jun. 2018.
- [19] R. Alcalá, J. Alcalá-fdez, J. Casillas, O. Cordon, and F. Herrera, "Local identification of prototypes for genetic learning of accurate TSK fuzzy rule-based systems," *Int. J. Intell. Syst.*, vol. 22, pp. 909–941, 2007.
- [20] P. Chang and C. Liu, "A TSK type fuzzy rule based system for stock price prediction," *Expert Syst. Appl.*, vol. 34, pp. 135–144, 2008.
- [21] J. Shell and S. Coupland, "Fuzzy transfer learning: Methodology and application," *Inf. Sci.*, vol. 293, pp. 59–79, 2015.
- [22] Z. Deng, Y. Jiang, K. S. Choi, F. L. Chung, and S. Wang, "Knowledge-leverage-based TSK fuzzy system modeling," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 8, pp. 1200–1212, Aug. 2013.
- [23] Z. Deng, Y. Jiang, L. Cao, and S. Wang, "Knowledge-leverage based TSK fuzzy system with improved knowledge transfer," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Beijing, China, 2014, pp. 178–185.
- [24] C. Yang, Z. Deng, K. Choi, and S. Wang, "Takagi–Sugeno–Kang transfer learning fuzzy logic system for the adaptive recognition of epileptic electroencephalogram signals," *IEEE Trans. Fuzzy Syst.*, vol. 24, no. 5, pp. 1079–1094, Oct. 2016.
- [25] H. Zuo, G. Zhang, W. Pedrycz, V. Behbood, and J. Lu, "Fuzzy regression transfer learning in Takagi–Sugeno fuzzy models," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 6, pp. 1795–1807, Dec. 2017.
- [26] H. Zuo, G. Zhang, J. Lu, and W. Pedrycz, "Fuzzy rule-based transfer learning for label space adaptation," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Naples, Italy, 2017, pp. 1–6.
- [27] Y. Jiang *et al.*, "Seizure classification from EEG signals using transfer learning, semi-supervised learning and TSK fuzzy system," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 12, pp. 2270–2284, Dec. 2017.
- [28] H. Zuo, J. Lu, G. Zhang, and F. Liu, "Fuzzy transfer learning using an infinite Gaussian mixture model and active learning," *IEEE Trans. Fuzzy Syst.*, vol. 27, no. 2, pp. 291–303, Feb. 2019.
- [29] H. Zuo, G. Zhang, W. Pedrycz, V. Behbood, and J. Lu, "Granular fuzzy regression domain adaptation in Takagi–Sugeno fuzzy models," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 2, pp. 847–858, Apr. 2018.
- [30] S. J. Pan, J. T. Kwok, and Q. Yang, "Transfer learning via dimensionality reduction," in *Proc. 23rd Int. Conf. Artif. Intell.*, 2008, vol. 2, pp. 677–682.
- [31] B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars, "Unsupervised visual domain adaptation using subspace alignment," in *Proc. IEEE Int. Conf. Comput. Vision*, Sydney, NSW, Australia, 2013, pp. 2960–2967.
- [32] R. Aljundi, R. Emonet, D. Muselet, and M. Sebban, "Landmarks-based kernelized subspace alignment for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 56–63.
- [33] L. Duan, I. W. Tsang, and D. Xu, "Domain transfer multiple kernel learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 3, pp. 465–479, Mar. 2012.
- [34] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, no. 1, pp. 116–132, Jan./Feb. 1985.
- [35] D. Dubois and H. Prade, "Fuzzy sets and systems: Theory and applications," *J. Oper. Res. Soc.*, vol. 33, pp. 198–198, 1997.
- [36] B. Rezaee and M. H. F. Zarandi, "Data-driven fuzzy modeling for Takagi–Sugeno–Kang fuzzy system," *Inf. Sci.*, vol. 180, pp. 241–255, 2010.
- [37] J. M. Mendel, "Fuzzy logic systems for engineering: A tutorial," *Proc. IEEE*, vol. 83, no. 3, pp. 345–377, Mar. 1995.
- [38] L. Xie, Z. Deng, P. Xu, K. S. Choi, and S. Wang, "Generalized hidden-mapping transductive transfer learning for recognition of epileptic electroencephalogram signals," *IEEE Trans. Cybern.*, vol. 49, no. 6, pp. 2200–2214, Jun. 2019.
- [39] Y. Chen, S. Song, S. Li, L. Yang, and C. Wu, "Domain space transfer extreme learning machine for domain adaptation," *IEEE Trans. Cybern.*, vol. 49, no. 5, pp. 1909–1922, May 2019.
- [40] B. Schölkopf, J. Platt, and T. Hofmann, "A kernel method for the two-sample problem," in *Proc. Conf. Adv. Neural Inf. Process. Syst.*, 2007, pp. 513–520.
- [41] T. Su and J. G. Dy, "In search of deterministic methods for initializing K-means and Gaussian mixture clustering," *Intell. Data Anal.*, vol. 11, pp. 319–338, 2007.
- [42] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," *Proc. 11th Eur. Conf. Comput. Vis.*, 2010, pp. 213–226.
- [43] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 2066–2073.
- [44] G. Griffin, A. Holub, and P. Perona, "Caltech-256 object category dataset," California Inst. Tech., Pasadena, CA, USA, Tech. Rep. 7694, 2007.
- [45] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Comput. Vis. Image Understand.*, vol. 110, pp. 346–359, 2008.
- [46] B. Chen, W. Lam, I. Tsang, and T. L. Wong, "Extracting discriminative concepts for domain adaptation in text mining," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, Paris, France, Jun./Jul. 2009, pp. 179–188.
- [47] M. Long, J. Wang, J. Sun, and P. S. Yu, "Domain invariant transfer kernel learning," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 6, pp. 1519–1532, Jun. 2015.
- [48] Q. V. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proc. 31st Int. Conf. Mach. Learn.*, 2014, pp. II-1188–II-1196.
- [49] S. Bird, E. Klein, and E. Loper, *Natural Language Processing With Python: Analyzing Text With the Natural Language Toolkit*. Newton, MA, USA: O'Reilly Media, 2009.
- [50] Z. Deng, K. Choi, Y. Jiang, and S. Wang, "Generalized hidden-mapping ridge regression, knowledge-leveraged inductive transfer learning for neural networks, fuzzy systems and kernel methods," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2585–2599, Dec. 2014.
- [51] S. Wu and M. J. Er, "Dynamic fuzzy neural networks—A novel approach to function approximation," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 30, no. 2, pp. 358–364, Apr. 2000.

- [52] P. Angelov and D. P. Filev, "An approach to online identification of Takagi–Sugeno fuzzy models," *IEEE Trans. Syst., Man, Cybern., B, Cybern.*, vol. 34, no. 1, pp. 484–498, Feb. 2004.
- [53] H. Rong, N. Sundararajan, G. Huang, and P. Saratchandran, "Sequential adaptive fuzzy inference system (SAFIS) for nonlinear system identification and prediction," *Fuzzy Sets Syst.*, vol. 157, pp. 1260–1275, 2006.
- [54] F. Liu, J. Lu, and G. Zhang, "Unsupervised heterogeneous domain adaptation via shared fuzzy equivalence relations," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 6, pp. 3555–3568, Dec. 2018.
- [55] Y. Luo, Y. Wen, T. Liu, and D. Tao, "Transferring knowledge fragments for learning distance metric from a heterogeneous domain," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 4, pp. 1013–1026, Apr. 2019.
- [56] X. He, "Locality preserving projections," in *Proc. Adv. Neural Inf. Process. Syst.*, 2002, vol. 16, pp. 186–197.



Peng Xu received the B.S. degree in computer science from Jiangnan University, Wuxi, China, in 2017. He is currently working toward the master's degree in software engineering with the School of Digital Media, Jiangnan University.

He has authored and coauthored several papers in international conferences and journals, including AAAI, IEEE TRANSACTIONS ON FUZZY SYSTEMS, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, and IEEE TRANSACTIONS ON NEURAL SYSTEMS & REHABILITATION ENGINEERING. His research

interests include computational intelligence, machine learning, interpretable artificial intelligence, fuzzy modeling, and their applications.



Zhaohong Deng (M'12–SM'14) received the B.S. degree in physics from Fuyang Normal College, Fuyang, China, in 2002, and the Ph.D. degree in information technology and engineering from Jiangnan University, Wuxi, China, in 2008.

He is currently a Professor with the School of Digital Media, Jiangnan University. He has visited the University of California–Davis and the Hong Kong Polytechnic University for more than two years. He has authored or coauthored more than 150 research papers in international/national journals. His current

research interests include uncertainty modeling, neuro-fuzzy systems, pattern recognition, and their applications.

Prof. Deng was an Associate Editor or Guest Editor for several international journals, such as IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTATIONAL INTELLIGENCE, *Neurocomputing*, and *PLoS One*.



Jun Wang (M'14) received the Ph.D. degree in pattern recognition and intelligence systems from the School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing, China, in 2011.

He is a Research Assistant with the Department of Computing, Hong Kong Polytechnic University, Hong Kong, and a Postdoctoral Research Fellow with the Department of Radiology and BRIC, School of Medicine, University of North Carolina at Chapel Hill, USA. He is currently an Associate Professor with the School of Communication and Information Engineering, Shanghai University, China. He has authored/coauthored more than 50 articles in international/national journals. His research interests include machine learning, fuzzy systems, and medical image classification.



Qun Zhang received the B.S. degree in English from Nanjing Normal University, Nanjing, China, in 1998, and the M.S. degree in food science from Jiangnan University, Wuxi, China, in 2007.

She is currently a Professor with the Library of Jiangnan University. Her research interests include information service, information education, and information technology's application in library.



Kup-Sze Choi (M'97) received the Ph.D. degree in computer science and engineering from the Chinese University of Hong Kong, Hong Kong SAR, The People's Republic of China, in 2004.

He is currently an Associate Professor with the School of Nursing, Hong Kong Polytechnic University, and the Director of the Centre for Smart Health. His research interests include virtual reality and artificial intelligence, and their applications in medicine and healthcare.



Shitong Wang received the M.S. degree in computer science from the Nanjing University of Aeronautics and Astronautics, China, in 1987.

He has authored/coauthored about 200 papers in international/national journals and has also authored seven books. His research interests include artificial intelligence, neuro-fuzzy systems, pattern recognition, and image processing.