Unsupervised Heterogeneous Domain Adaptation via Shared Fuzzy Equivalence Relations

Feng Liu, *Student Member, IEEE*, Jie Lu, *Fellow, IEEE*, and Guangquan Zhang

***Abstract*— Unsupervised domain adaptation (UDA) aims to recognize newly emerged patterns in target domains, which may be unlabeled, by leveraging knowledge from patterns learnt from source domains. However, existing UDA models and algorithms still suffer from heterogeneous domains, known as the heterogeneous unsupervised domain adaptation (HeUDA) issue. To address this issue, this paper presents a novel HeUDA model via *n*-dimensional fuzzy geometry and fuzzy equivalence relations, called F-HeUDA. The *n*-dimensional fuzzy geometry is used to propose a metric to measure the similarity between features on one domain. Then, based on this metric, shared fuzzy equivalence relations (SFER) is proposed. The SFER can allow two domains to use the same α to get the same number of clustering categories. Through these clustering categories, knowledge from the heterogeneous source domain can be transferred to the unlabeled target domain. Different to existing HeUDA models, the proposed F-HeUDA model does not need that two domains must have the same number of instances. As a result, the proposed model has a better ability to handle the issue of small datasets. Experiments distributed across four real datasets were conducted to validate the proposed model. This testing regime demonstrates that the proposed model outperforms the state-of-the-art models, especially when the target domain has very few instances.**

***Index Terms*— Transfer learning, domain adaptation, fuzzy relations, machine learning**

I. Introduction

WHAT makes the human learning process advanced is our ability to transfer knowledge from experienced situations to a newly emerged one. This is the ability which is needed by an artificial intelligent model in order to, for example: 1) predict the demand for a new product using the knowledge of existing products; 2) diagnose a newly discovered cancer using knowledge of existing cancers; and 3) to assess the credit of a foreigner using existing national assessment systems. Artificial intelligence researchers first developed models which were trained by a training set and were then applied to predict the labels of instances of the testing set. This type of model, called the traditional machine learning model, has the ability to transfer knowledge from the training set to the testing set when these two sets have the same features. However, traditional machine learning models have unsatisfactory results when there is divergence between two sets (such as the divergence of the distributions of two sets). So, transfer learning models have been proposed to address this problem by minimizing the divergence between two sets [1], [2] and the training set and the testing set were extended to more general concepts: source domains and target domains [3]–[5].

Subsequently, researchers began to consider how to leverage the knowledge obtained from a source domain to help predict the labels in a target domain where the two domains have different feature spaces. Until now, transfer learning models have attracted a large amount attention and made fast progress in both theory and practice. Examples include using classified French documents to help classify English documents [6]; building recognition models capable of recognising novel visual categories without labelled training samples [7]; detecting a user’s current location based on previously collected WiFi data [3] and leveraging the large number of labeled simple actions to recognize complex human actions [8]. As the main type of transfer learning models, domain adaptation models have demonstrated great success in recent years [9], [10]. Domain adaptation models aim to transfer knowledge between two domains which perform similar tasks, such as classifying news documents [6], recognizing similar objects [4] and predicting the value of owner-occupied homes [11]. There are two major categories of domain adaptation models: homogeneous and heterogeneous.

Homogeneous domain adaptation models were proposed to minimize the divergence between distributions of feature spaces of two domains. At first, researchers focused on the issue that two domains have the same feature space but different distributions. Then, researchers realized that homogeneous domain adaptation models can address more general issues such as when two domains have different features (the number of features of two domains is the same). When there are labeled instances in the target domain, the representative models include adaptive support vector machines [12], projective model transfer SVM [13], Bayesian co-training [14] and max-

Manuscript received September 18, 2017; revised February 14, 2018; accepted May 3, 2018. The work presented in this paper was supported by the Australian Research Council (ARC) under discovery grant DP170101632.

Feng Liu, Guangquan Zhang and Jie Lu are with the Decision Systems & e-Service Intelligence Lab, Centre for Artificial Intelligence, Faculty of Engineering and Information Technology, University of Technology Sydney, Australia (e-mail: Feng.Liu-2@student.uts.edu.au/Feng.Liu.1990@ieee.org; Guangquan.Zhang@uts.edu.au; Jie.Lu@uts.edu.au).

margin domain transforms [15]. These models need labeled instances in the target domain and are called homogeneous (semi) supervised domain adaptation models. When there is no labeled instance in the target domain, domain adaptation models are called homogeneous unsupervised domain adaptation (HoUDA) models. The representative HoUDA models include transfer component analysis [3], geodesic flow kernel (GFK) [4], [16], information-theoretical learning [17] and transfer deep network [18]. Most HoUDA models aim to minimize the divergence between distributions of two feature spaces. Frequently used mathematical tools are reproducing kernel Hilbert space [3], [9], [19], Grassmann manifold [4], [20] and graph matching [10]. Compared to homogeneous domain adaptation models, heterogeneous domain adaptation models have more general application scenarios because they can transfer knowledge from heterogeneous domains to the target domain. When the target domain has labeled instances, representative heterogeneous domain adaptation models include heterogeneous feature augmentation [21], asymmetric regularized cross-domain transformation [22], heterogeneous spectral mapping [23], manifold alignment-based models [24], semi-supervised kernel matching for domain adaptation [6], and the DASH-N model [25]. These models can also be regarded as heterogeneous (semi) supervised domain adaptation models. In a situation where there is no labeled instance in the target domain (the most challenging task in the field of domain adaptation), there are rare models which exist because current models have two bottlenecks: 1) measuring the distance between two heterogeneous feature spaces and 2) theoretically avoiding negative transfer. Although kernel canonical correlation analysis (KCCA) [26] was proposed as a heterogeneous unsupervised domain adaptation (HeUDA) model, it needs two domains which have paired instances.

To propose an effective HeUDA model, we successfully designed Grassmann linear monotonic maps geodesic flow kernel (GLG) in our previous work [27], which solves the two aforementioned bottlenecks and has satisfactory classification results on three real applications. However, no matter which HeUDA model (KCCA or GLG) is used, it does not work well when the target domain is a small dataset due to the limitations of CCA and the Grassmann manifold (both of which need two domains with the same number of instances). This will limit the amount of knowledge from the source domain which is transferred to the target domain. For example, if we have a source domain containing 10,000 instances, we need to label instances of the target domain which only have 50 instances. If we use the KCCA or GLG model, we can only select 50 instances in the source domain and transfer the knowledge from these 50 selected instances to the target domain. This means we waste 99.5% of the information of the source domain, which is not acceptable. Thus, to address this problem, this paper applies the n-dimensional fuzzy geometry and the fuzzy equivalence relations to transfer all of the knowledge from a heterogeneous domain to the target domain, where two domains have a different number of instances.

Fuzzy technology plays an important role in the field of artificial intelligence because it can deal with the uncertainty of the dataset and give a reasonable explanation of the dataset itself. In the field of domain adaptation, fuzzy rules and Takagi – Sugeno models are also frequently used to transfer knowledge across domains [5], [11], [28]–[30] because they can extract general fuzzy representations of two domains and illustrate how knowledge is transferred across domains. Deng et al. proposed a series of novel models to effectively transfer knowledge across domains using Mamdani-Larsen-type fuzzy system, Takagi-Sugeno-Kang-type fuzzy system, including knowledge-leverage-based Mamdani-Larsen-type fuzzy system (KL-ML-FS) [31], knowledge-leverage-based Takagi-Sugeno-Kang-type fuzzy system (KL-TSK-FS) [32] and enhanced KL-TSK-FS (EKL-TSK-FS) [33]. They also proposed a novel clustering model, transfer prototype - based fuzzy clustering (TPFC) [34] and a novel regression model, transfer generalized hidden-mapping ridge regression (TGHRR) [35]. Sun et al. proposed a granular transfer learning with type-2 fuzzy hidden Markov model (GT2HMM) [36] and introduced the granular computing into the processing of contextual uncertainty for transfer learning. These fuzzy transfer models demonstrate that fuzzy technologies make knowledge transfer more effectively. Based on these advantages of fuzzy technology, we first propose a new metric $\mathcal{D}$ on an n-dimensional fuzzy space $\mathcal{F}(\mathbb{R}^n)$ where each feature of a domain is regarded as a fuzzy vector. This new metric contains fuzzy degrees of fuzzy vectors and it is proved that $\left(\mathcal{D},\mathcal{F}(\mathbb{R}^n)\right)$ is a metric space. Then, we use this metric to measure the similarity of two fuzzy vectors and build the fuzzy equivalence relations matrix of each domain. In a traditional fuzzy equivalence relations matrix, we can use an α to cluster these fuzzy vectors (representing features) into several categories and these categories are regarded as more general fuzzy representations. Motivated by this, we propose the shared fuzzy equivalence relations (SFER), which allows two fuzzy equivalence matrixes of two domains to share the same α. Compared to the traditional fuzzy equivalence relations, the SFER can guarantee that fuzzy equivalence relations matrixes of two domains can have the same number of clustering categories with the same α (traditional fuzzy equivalence relations cannot guarantee this). Eventually, with the help of the SFER, we can transfer knowledge from the source domain to the target domain via these clustering categories.

The main contributions of this paper can be summarized as follows.

1) This paper proposes a novel F-HeUDA model, adopting n-dimensional fuzzy geometry and fuzzy equivalence relations to address heterogeneous domain adaptation issues. Both fuzzy technologies successfully overcome the drawbacks of CCA and the Granssmann manifold (two domains must have the same number of instances). As a result, the proposed model can transfer more knowledge from a source domain to a target domain than KCCA and GLG models when there are very few instances in the target domain.

2) Two important properties of fuzzy equivalence relations are discovered and proved in this paper, which are the theoretical guarantees of the SFER model and key parts of the proposed model. Based on both properties, it can be guaranteed that fuzzy equivalence relations matrixes of two domains can

have the same number of clustering categories with the same and proper α.

3) This is the first time that *n*-dimensional fuzzy geometry and fuzzy equivalence relations have been applied to address the issue of domain adaptation and the proposed model performs well in real applications.

4) The F-HeUDA model provides an "α-cut" decision making pattern for decision makers. In the proposed model, we propose a default method by which to automatically select α for different tasks which works well in four real applications. Furthermore, decision makers can still easily select the value of α based on their own experience and requirements. This extends the application scenario of the proposed model.

The remainder of this paper is organized as follows. Section II briefly introduces the *n*-dimensional fuzzy geometry and the fuzzy equivalence relations. Section III proposes a way to measure the similarity of two fuzzy vectors. In Section IV, we prove two important properties of the fuzzy equivalence relations and propose the F-HeUDA model. Section V describes the use of four real datasets to test the performance of the proposed models and benchmarks, and demonstrates the convergence of the proposed learning algorithm for the SFER, and shows the "α-cut" decision making pattern of the proposed model. Finally, Section VI concludes the paper and outlines future studies.

II. PRELIMINARY

This section introduces the two basic concepts used in this paper, which are *n*-dimensional fuzzy geometry (*n*-D FG) and fuzzy equivalence relations.

A. N-dimensional fuzzy geometry

The geometry properties of fuzzy sets have been extensively researched in various aspects such as fuzzy point, fuzzy line and fuzzy circle [37]–[41]. The *n*-D FG theory is developed to provide an effective way to analyze and compute fuzzy information in a geometry form [42]. In this subsection, several definitions are introduced to explain the *n*-D fuzzy vector, the core element in an *n*-D FG. Without loss of generality, this paper uses capital or small letters with a bar to represent the fuzzy points or fuzzy subsets of $\mathbb{R}^n$. The membership function of a fuzzy set $\bar{\mathrm{A}}$ is denoted by $\mu(x|\bar{\mathrm{A}})$, $x \in \mathbb{R}^n$, with $\mu(x|\bar{\mathrm{A}}) \subseteq [0,1]$. First, the fuzzy number is defined as follows.

Definition 1 [43]: A fuzzy set $\bar{\mathrm{S}}$ of $\mathbb{R}$ is called a fuzzy real number (fuzzy number in the rest of this paper) if its membership function μ satisfies the following properties.

1. $\mu(x|\bar{\mathrm{S}})$ = 1 is upper semi-continuous in *x*.

2. $\mu(x|\bar{\mathrm{S}})$ = 1 for *x* outside some interval $[c,d]$.

3. For some real numbers with $c \le a \le b \le d$, $\mu(x|\bar{\mathrm{S}})$ is monotonically increasing in $[c,a]$, monotonically decreasing in $[b,d]$, and $\mu(x|\bar{\mathrm{S}})$ = 1 for $x \in [a,b]$.

Then, we can give the definition of the fuzzy vector at an *n*-D vector as follows.

Definition 2 [43]: A fuzzy set $\bar{\mathrm{A}}(a_1,a_2,\dots,a_n)$ of $\mathbb{R}^n$ is called a fuzzy vector at $\mathrm{A} = (a_1,a_2,\dots,a_n) \in \mathbb{R}^n$ if its membership function μ satisfies the following properties.

1. $\mu\big((x_1,x_2,\dots,x_n)\big|\bar{\mathrm{A}}(a_1,a_2,\dots,a_n)\big)$ = 1 is upper semi-continuous in $\boldsymbol{x}$ = $(x_1,x_2,\dots,x_n) \in \mathbb{R}^n$.

2. $\mu\big((x_1,x_2,\dots,x_n)\big|\bar{\mathrm{A}}(a_1,a_2,\dots,a_n)\big)$ = 1 if and only if $(x_1,x_2,\dots,x_n) = (a_1,a_2,\dots,a_n)$.

3. $\bar{\mathrm{A}}(\alpha) = \big\{\boldsymbol{x}\big|\mu\big(\boldsymbol{x}\big|\bar{\mathrm{A}}(a_1,a_2,\dots,a_n)\big) = \alpha, \boldsymbol{x} \in \mathbb{R}^n\big\}$ is a compact convex subset of $\mathbb{R}^n$ for all α in [0, 1].

The fuzzy vector is the basic element for researching properties of the *n*-D FG space and the set of all *n*-D fuzzy vectors is denoted by $F(\mathbb{R}^n)$. The third property of *n*-D fuzzy vectors implies that $F(\mathbb{R}^n)$ can be connected with $\mathbb{R}^n$ using the membership α.

This paper uses the triangular membership function to construct the membership function of each *n*-D fuzzy vector in $F(\mathbb{R}^n)$. The detailed form is introduced in section III.

B. Fuzzy equivalence relations

Fuzzy equivalence relations are first mentioned in [44] (Zadeh referred to fuzzy equivalence relations as similarity relations in [44]) and were studied as a way to measure the similarity among fuzzy sets. Based on the fuzzy equivalence relations, fuzzy equivalence classes can be obtained, which provides a powerful way to analyze the fuzzy partitions. Then, to construct the fuzzy equivalence relations for general fuzzy sets, the max–min operator is proposed to construct max–min transitive closure which is a fuzzy equivalence relations [45]. This section briefly reviews how to generate a fuzzy equivalence relations for fuzzy sets and how to use the fuzzy equivalence relations to partition fuzzy sets.

First, the definition of a fuzzy relation is given as follows.

Definition 3: Given *N* fuzzy sets, $\overline{\mathrm{A}_1}, \overline{\mathrm{A}_2}, \dots, \overline{\mathrm{A}_N}$, an operator $R: (\bar{\mathrm{A}}_i, \bar{\mathrm{A}}_j) \mapsto [0,1]$ is a fuzzy relation on $\overline{\mathrm{A}_1}, \overline{\mathrm{A}_2}, \dots, \overline{\mathrm{A}_N}$ if the following properties are satisfied.

1) $R(\bar{\mathrm{A}}_i, \bar{\mathrm{A}}_j) = 1 \; \forall \bar{\mathrm{A}}_i$ (reflexivity),

2) $R(\bar{\mathrm{A}}_i, \bar{\mathrm{A}}_j) = R(\bar{\mathrm{A}}_j, \bar{\mathrm{A}}_i), \forall \bar{\mathrm{A}}_i, \bar{\mathrm{A}}_j$ (symmetry).

It is obvious that the fuzzy relation R on $\overline{\mathrm{A}_1}, \overline{\mathrm{A}_2}, \dots, \overline{\mathrm{A}_N}$ can be expressed by a *N*-by-*N* matrix $R^M = (r_{ij}), r_{ij} = R(\bar{\mathrm{A}}_i, \bar{\mathrm{A}}_j)$. Then the max–min operator $\circ$ is defined for two fuzzy relations matrixes R_a^M and R_b^M.

$$(R_a^M \circ R_b^M)_{ij} = \bigvee_{k=1}^{N} \left(r_{ik}^{(a)} \wedge r_{kj}^{(b)}\right), \qquad (1)$$

where $r_{ik}^{(a)}$ is the element of R_a^M and $r_{kj}^{(b)}$ is the element of R_b^M, "$\wedge$" represents the minimize, "$\vee$" represents the maximize. It is clear that $R_a^M \circ R_b^M$ is also a fuzzy relations matrix and $r_{ij}^{(a)} \le (R_a^M \circ R_b^M)_{ij}$ and $r_{ij}^{(b)} \le (R_a^M \circ R_b^M)_{ij}$.

Next, the fuzzy equivalence relation is defined as follows.

Definition 4: Given *N* fuzzy sets, $\overline{\mathrm{A}_1}, \overline{\mathrm{A}_2}, \dots, \overline{\mathrm{A}_N}$, an operator $R: (\bar{\mathrm{A}}_i, \bar{\mathrm{A}}_j) \mapsto [0,1]$ is a fuzzy equivalence relation on $\overline{\mathrm{A}_1}, \overline{\mathrm{A}_2}, \dots, \overline{\mathrm{A}_N}$ if the following properties are satisfied.

1) $R(\bar{\mathrm{A}}_i, \bar{\mathrm{A}}_j) = 1, \forall \bar{\mathrm{A}}_i$ (reflexivity),

2) $R(\bar{\mathrm{A}}_i, \bar{\mathrm{A}}_j) = R(\bar{\mathrm{A}}_j, \bar{\mathrm{A}}_i), \forall \bar{\mathrm{A}}_i, \bar{\mathrm{A}}_j$ (symmetry),

3) $R_{(2)}^M = R_{(1)}^M \circ R_{(1)}^M$ (transitivity).

where R^M is the fuzzy relation matrix on R and $\circ$ is the max-min operator mentioned above.

Compared to fuzzy equivalence relations, fuzzy relations are much easier to obtain because fuzzy relations do not require transitivity. This leads researchers to find a way to construct the fuzzy equivalence relations based on the fuzzy relations and the max-min operator. The following theorem is provided to show that the max–min transitive closure R_T of a fuzzy relation R is a fuzzy equivalence relation.

Theorem 1: Given a fuzzy relation R on $\overline{\mathrm{A}}_1, \overline{\mathrm{A}}_2, \dots, \overline{\mathrm{A}}_N$, there must be a finite $m \in \mathbb{Z}$ and an operator R_T satisfies the following conditions.

1) $R_T^M = R_{(m)}^M = \underbrace{R^M \circ R^M \circ \dots \circ R^M}_{m}$.

2) $R_T^M = R_T^M \circ R_T^M$

where R^M is the fuzzy relations matrix of R and R_T^M is the fuzzy relations matrix of R_T. The operator R_T is called the max-min transitive closure of R.

Proof: Based on Theorem 5.1 and Theorem 5.2 in [46], R_T satisfies the following equations.

$$R_T^M = \bigvee_{k=1}^{\infty} R_{(k)}^M,$$

$$R_T^M = R_{(N-1)}^M.$$

Thus, $m = N-1$ (condition 1 is satisfied) and we have

$$\begin{aligned} R_T^M \circ R_T^M &= \bigvee_{k=1}^{\infty} R_{(k)}^M \circ \bigvee_{l=1}^{\infty} R_{(l)}^M \\ &= \bigvee_{k=1}^{\infty} \bigvee_{l=1}^{\infty} R_{(l)}^M \circ R_{(k)}^M \\ &= \bigvee_{l,k=1}^{\infty} R_{(l+k)}^M = R_T^M \end{aligned}$$

thus, condition 2 is satisfied. □

Theorem 1 provides a way to construct fuzzy equivalence relations through fuzzy relations. With the constructed fuzzy equivalence relations, we can use α-cut of R_T^M to cluster $\overline{\mathrm{A}}_1, \overline{\mathrm{A}}_2, \dots, \overline{\mathrm{A}}_N$. Specifically, the matrix of α-cut of R_T^M can be expressed by the following term.

$$\left(R_T^M(\alpha)\right)_{ij} = \begin{cases} 1, if\ (R_T^M)_{ij} \geq \alpha \\ 0, if\ (R_T^M)_{ij} < \alpha \end{cases}. \qquad (2)$$

$R_T^M(\alpha)$ is a binary fuzzy equivalence relation matrix. Fuzzy sets that have the same corresponding rows of $R_T^M(\alpha)$ can be regarded as the same cluster. The selection of α is a decision-making process, and users can choose α based on their own requirements.

Traditional fuzzy equivalence relations are only for one type of fuzzy set, such as the set $\overline{S_1} = \{\overline{\mathrm{A}_1}, \overline{\mathrm{A}_2}, \dots, \overline{\mathrm{A}_{N_1}}\}, \overline{\mathrm{A}}_\iota \in F(\mathbb{R}^{n_1})$. However, for the HeUDA problem, there is always another set $\overline{S_2} = \{\overline{\mathrm{B}_1}, \overline{\mathrm{B}_2}, \dots, \overline{\mathrm{B}_{N_2}}\}$, $\overline{\mathrm{B}}_\iota \in F(\mathbb{R}^{n_2})$ and $n_1 \neq n_2$, $N_1 \neq N_2$. In general, R_T^M of S_1 and S_2 are different and cannot get the same number of clusters through sharing the same α, which is the main obstacle when dealing with the HeUDA problem using fuzzy relations. In order to let R_T^M of S_1 and S_2 share the same α, this paper designs a new shared fuzzy equivalence relation (SFER) which is based on two sets. Some new properties of traditional fuzzy equivalence are first discussed in Section IV, then, SFER is detailed based on these new properties.

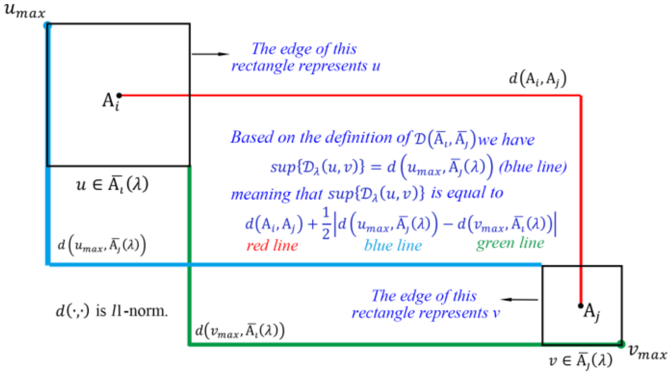

u_{max}
The edge of this rectangle represents u
$d(\mathrm{A}_i, \mathrm{A}_j)$
A_i
Based on the definition of $\mathcal{D}(\overline{\mathrm{A}}_\iota, \overline{\mathrm{A}}_j)$ we have
$sup\{\mathcal{D}_\lambda(u,v)\} = d\left(u_{max}, \overline{\mathrm{A}}_j(\lambda)\right)$ (blue line)
meaning that $sup\{\mathcal{D}_\lambda(u,v)\}$ is equal to
$d(\mathrm{A}_i, \mathrm{A}_j) + \frac{1}{2}\left|d\left(u_{max}, \overline{\mathrm{A}}_j(\lambda)\right) - d\left(v_{max}, \overline{\mathrm{A}}_\iota(\lambda)\right)\right|$
red line
blue line
green line
$u \in \overline{\mathrm{A}}_\iota(\lambda)$
$d\left(u_{max}, \overline{\mathrm{A}}_j(\lambda)\right)$
$d(\cdot,\cdot)$ is $l1$-norm.
$d\left(v_{max}, \overline{\mathrm{A}}_\iota(\lambda)\right)$
The edge of this rectangle represents v
A_j
$v \in \overline{\mathrm{A}}_j(\lambda)$
v_{max}

Fig. 1. Relationships among $sup\{\mathcal{D}_\lambda(u,v) \colon \mathcal{D}_\lambda(u,v) \in \Omega(\lambda)\}$, $d\left(u, \overline{\mathrm{A}}_j(\lambda)\right)$, $d\left(v, \overline{\mathrm{A}}_i(\lambda)\right)$ and $d(\mathrm{A}_i, \mathrm{A}_j)$

III. SIMILARITY OF N-DIMENSIONAL FUZZY VECTORS

In this section, we propose a new similarity between two fuzzy vectors in the *n*-D FG. The first subsection describes a new metric on *n*-D FG and proves its correctness. The second subsection proposes the new similarity based on the new metric defined in subsection A.

A. Metric on n-D FG

To measure the distance between two fuzzy vectors in the *n*-D FG, researchers defined several metrics, comprising two main types: fuzzy metrics [40], [47] and de-fuzzy metrics [42]. Although the literature [42] proposed a de-fuzzy metric based on the fuzzy metric defined in [40], this de-fuzzy metric cannot satisfy the second condition of a metric space. In this section, a proper de-fuzzy metric $\mathcal{D}$ is proposed and we prove that $(F(\mathbb{R}^n), \mathcal{D})$ is a metric space. First, the detailed expression of a fuzzy vector $\overline{\mathrm{A}}_i(a_{i1}, a_{i2}, \dots, a_{in}) \in F(\mathbb{R}^n)$ (with the triangular membership function) is given as follows: for each $\bar{a}_{ij} \in F(\mathbb{R})$, its membership function is

$$\mu_{ij}\left(x \middle| \bar{a}_{ij}\right) = \begin{cases} 0, & \forall x < a_{ij} - \rho_i \\ 1 - \dfrac{\left|x - a_{ij}\right|}{\rho_i}, & \forall \left|x - a_{ij}\right| \leq \rho_i \\ 0, & \forall x > a_{ij} + \rho_i \end{cases}, x \in \mathbb{R}, \qquad (3)$$

Based on the $\mu_{ij}\left(x \middle| \bar{a}_{ij}\right)$, $\mu_i(\boldsymbol{x}|\overline{\mathrm{A}}_i)$ is expressed by the following term ($\boldsymbol{x} = (x_1, x_2, \dots, x_n)$).

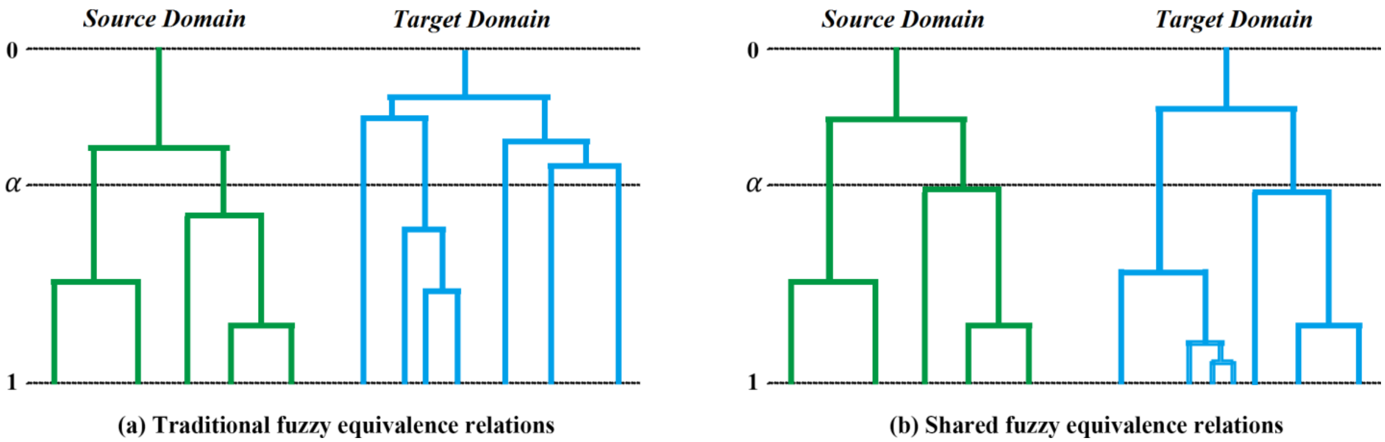
$$\mu_i(\boldsymbol{x}|\overline{\mathrm{A}}_i) = \begin{cases} 0, & if\ \exists x_j,\ \ x_j < a_{ij} - \rho_i \\ 1 - \dfrac{\left\|\boldsymbol{x} - a_{ij}\right\|_1}{n\rho_i}, & if\ \forall x_j, \left|x_j - a_{ij}\right| \leq \rho_i \\ 0, & if\ \exists x_j,\ \ x_j > a_{ij} + \rho_i \end{cases}, \boldsymbol{x} \in \mathbb{R}^{\boldsymbol{n}}, \qquad (4)$$

Then, we define a new metric to measure the distance between two fuzzy vectors.

Definition 5: Given two fuzzy vectors $\overline{\mathrm{A}}_i \in F(\mathbb{R}^n)$ and $\overline{\mathrm{A}}_j \in F(\mathbb{R}^n)$, the new metric between $\overline{\mathrm{A}}_i$ and $\overline{\mathrm{A}}_j$ is defined by the map $\mathcal{D}\colon F(\mathbb{R}^n) \times F(\mathbb{R}^n) \to [0, +\infty)$:

$$\mathcal{D}\left(\overline{\mathrm{A}}_i, \overline{\mathrm{A}}_j\right) = \frac{1}{n}\int_0^1 sup\{\mathcal{D}_\lambda(u,v) \colon \mathcal{D}_\lambda(u,v) \in \Omega(\lambda)\} d\lambda,$$

$$\Omega(\lambda) = \left\{d\left(u, \overline{\mathrm{A}}_j(\lambda)\right)\right\} \cup \left\{d\left(v, \overline{\mathrm{A}}_i(\lambda)\right)\right\},$$



(a) Traditional fuzzy equivalence relations

(b) Shared fuzzy equivalence relations

Fig. 2. Traditional fuzzy equivalence relations v.s. SFER. In subfigure (a), two domains clearly cannot use the same α to obtain the same number of clusters. But in the proposed model, SFER, two domains have a much bigger probability of using the same α to obtain the same number of clusters.

where $u \in \overline{\mathrm{A}}_i(\lambda), v \in \overline{\mathrm{A}}_j(\lambda)$ and the first part of $\Omega(\lambda)$ collects L_1 distances between each u and $\overline{\mathrm{A}}_j(\lambda)$ ($d(u,\overline{\mathrm{A}}_j(\lambda)) = min\{d(u,v), v \in \overline{\mathrm{A}}_j(\lambda)\}$ means the minimum L_1 distances between u and all elements in $\overline{\mathrm{A}}_j(\lambda)$), and the second part of $\Omega(\lambda)$ collects L_1 distances between v and $\overline{\mathrm{A}}_i(\lambda)$ ($d\big(v,\overline{\mathrm{A}}_i(\lambda)\big) = min\ \{d(v,u), u \in \overline{\mathrm{A}}_i(\lambda)\}\}$ means the minimum L_1 distances between v and all elements in $\overline{\mathrm{A}}_i(\lambda)$), and $d(u,v)$ represents the L_1 distance (ℓ_1-norm) between two n-dimension vector (u and v).

Remark 1: *To measure the distance between two fuzzy vectors is a key to define the fuzzy relation between them. Thus, we first propose a new measurement represented in Definition 5. $\mathcal{D}\big(\overline{\mathrm{A}}_i,\overline{\mathrm{A}}_j\big)$ is the longest distance among 1) distances between v and $\overline{\mathrm{A}}_i$ and 2) distances between u and $\overline{\mathrm{A}}_j$.*

Figure 1 shows the meaning of $\mathcal{D}\big(\overline{\mathrm{A}}_i,\overline{\mathrm{A}}_j\big)$ and indicates that the following equation exists.

$$\mathcal{D}\big(\overline{\mathrm{A}}_i,\overline{\mathrm{A}}_j\big) = \frac{1}{n}\int_0^1 d\big(\mathrm{A}_i,\mathrm{A}_j\big) + \frac{1}{2}\left|d\left(u,\overline{\mathrm{A}}_j(\lambda)\right) - d\big(v,\overline{\mathrm{A}}_i(\lambda)\big)\right| d\lambda. \quad (5)$$

Based on $\mu_i(\boldsymbol{x}|\overline{\mathrm{A}}_i)$, we derive the following equations:

$$\begin{aligned}\mathcal{D}\big(\overline{\mathrm{A}}_i,\overline{\mathrm{A}}_j\big) &= \tfrac{1}{n}d\big(\mathrm{A}_i,\mathrm{A}_j\big) + \tfrac{1}{2}\textstyle\int_0^1\big|(1-\lambda)\rho_i - (1-\lambda)\rho_j\big|\, d\lambda \\ &= \frac{1}{n}d\big(\mathrm{A}_i,\mathrm{A}_j\big) + \frac{1}{2}\big|\rho_i - \rho_j\big|\int_0^1 (1-\lambda)\, d\lambda \\ &= \frac{1}{n}d\big(\mathrm{A}_i,\mathrm{A}_j\big) + \frac{1}{4}\big|\rho_i - \rho_j\big|. \quad (6)\end{aligned}$$

Theorem 2: $(F(\mathbb{R}^n),\mathcal{D})$ is a metric space.

Proof: To prove this theorem, we need to prove $\mathcal{D}$ can satisfy the following conditions for $\forall\overline{\mathrm{A}}_i,\overline{\mathrm{A}}_j$ and $\overline{\mathrm{A}}_k \in F(\mathbb{R}^n)$.

1) $\mathcal{D}\big(\overline{\mathrm{A}}_i,\overline{\mathrm{A}}_j\big) \geq 0$;

2) $\mathcal{D}\big(\overline{\mathrm{A}}_i,\overline{\mathrm{A}}_j\big) = 0$ if and only if $\overline{\mathrm{A}}_i = \overline{\mathrm{A}}_j$;

3) $\mathcal{D}\big(\overline{\mathrm{A}}_i,\overline{\mathrm{A}}_j\big) = \mathcal{D}\big(\overline{\mathrm{A}}_j,\overline{\mathrm{A}}_i\big)$;

4) $\mathcal{D}\big(\overline{\mathrm{A}}_i,\overline{\mathrm{A}}_j\big) \leq \mathcal{D}\big(\overline{\mathrm{A}}_i,\overline{\mathrm{A}}_k\big) + \mathcal{D}\big(\overline{\mathrm{A}}_k,\overline{\mathrm{A}}_j\big)$;

where $\overline{\mathrm{A}}_i = \overline{\mathrm{A}}_j$ means that $A_i = A_j$ and $\rho_i = \rho_j$.

From the definition of $\mathcal{D}$, it is clear that conditions 1) and 3) are achieved. Because $d(\cdot,\cdot)$ represents the L_1 distance, condition 2) can be satisfied. For condition 4), we have

$$d\big(\mathrm{A}_i,\mathrm{A}_j\big) \leq d\big(\mathrm{A}_i,\mathrm{A}_k\big) + d\big(\mathrm{A}_k,\mathrm{A}_j\big)$$

and

$$\big|\rho_i - \rho_j\big| \leq \big|\rho_i - \rho_k\big| + \big|\rho_k - \rho_j\big|.$$

Thus, condition 4) is satisfied. □

Theorem 2 shows the correctness of definition 5 and gives a new perspective on the measurement of two fuzzy vectors.

B. Similarity of n-D fuzzy vectors

The metric $\mathcal{D}$ can map two fuzzy vectors to a real positive number, which cannot be directly used as a fuzzy relation (Definition 3). Thus, this subsection transforms $\mathcal{D}$ into a fuzzy relation $R_{\mathcal{D}}$ as follows.

Lemma 1: Given two fuzzy vectors where $\overline{\mathrm{A}}_i \in F(\mathbb{R}^n)$ and $\overline{\mathrm{A}}_j \in F(\mathbb{R}^n)$, if an operator $R_{\mathcal{D}}$: $\big(\overline{\mathrm{A}}_i,\overline{\mathrm{A}}_j\big) \mapsto [0,1]$ derived by $\mathcal{D}$ is defined as follows,

$$R_{\mathcal{D}}\big(\overline{\mathrm{A}}_i,\overline{\mathrm{A}}_j\big) = e^{-\frac{\mathcal{D}(\overline{\mathrm{A}}_i,\overline{\mathrm{A}}_j)}{2\sigma^2}}, \quad (7)$$

then $R_{\mathcal{D}}$ is a fuzzy relation.

Proof: Because $f(x) = e^{\frac{-x}{2\sigma^2}}$ is monotonic when $x \in [0,+\infty)$ and $(F(\mathbb{R}^n),\mathcal{D})$ is a metric space, we have

$$R_{\mathcal{D}}\big(\overline{\mathrm{A}}_i,\overline{\mathrm{A}}_j\big) = R_{\mathcal{D}}\big(\overline{\mathrm{A}}_j,\overline{\mathrm{A}}_i\big). \quad (8)$$

Also, we have

$$R_{\mathcal{D}}\big(\overline{\mathrm{A}}_i,\overline{\mathrm{A}}_i\big) = e^{-\frac{0}{2\sigma^2}} = 1. \quad (9)$$

So, based on Definition 3 and Eqs. (8) and (9), the operator $R_{\mathcal{D}}$ satisfies reflexivity and symmetry, meaning that $R_{\mathcal{D}}$ is a fuzzy relation. □

Based on the $R_{\mathcal{D}}$, we can obtain the fuzzy relations matrix of $R_{\mathcal{D}}$ and denote this matrix by $R_{\mathcal{D}}^M$, where $R_{\mathcal{D}}^M$ is a squared matrix and $(R_{\mathcal{D}}^M)_{ij} = R_{\mathcal{D}}\big(\overline{\mathrm{A}}_i,\overline{\mathrm{A}}_j\big), i,j = 1,2,\ldots,N$. Furthermore, the max-min transitive closure of $R_{\mathcal{D}}$ is denoted by $R_{T\mathcal{D}}$ and the $R_{T\mathcal{D}}^M$ is the fuzzy relations matrix of $R_{T\mathcal{D}}$, where $R_{T\mathcal{D}}^M$ is a squared matrix and $(R_{T\mathcal{D}}^M)_{ij} = R_{T\mathcal{D}}\big(\overline{\mathrm{A}}_i,\overline{\mathrm{A}}_j\big), , i,j = 1,2,\ldots,N$. Based on Theorem 1, we also know that $R_{T\mathcal{D}}^M = \underbrace{R_{\mathcal{D}}^M \circ R_{\mathcal{D}}^M \circ \ldots \circ R_{\mathcal{D}}^M}_{N-1}$.

IV. HeUDA via Shared fuzzy equivalence relations

Before introducing the proposed model, the aim of a HeUDA model is formally demonstrated as follows. Considering two feature sets of source and target domains: $S_1 = \{\mathrm{A}_1,\mathrm{A}_2,\ldots,\mathrm{A}_{N_1}\}$, $\mathrm{A}_i \in \mathbb{R}^{n_1}$, $S_2 = \{\mathrm{B}_1,\mathrm{B}_2,\ldots,\mathrm{B}_{N_2}\}$, $\mathrm{B}_i \in \mathbb{R}^{n_2}$ and $n_1 \neq n_2$, $N_1 \neq N_2$, the HeUDA models aim to find a common feature set $S_c = \{\mathrm{C}_1,\mathrm{C}_2,\ldots,\mathrm{C}_{N_c}\}$, $\mathrm{C}_i \in \mathbb{R}^{n_c}$, $n_c = n_1 + n_2$, and use the labeled information of S_c (knowledge from the source domain) to label the unlabeled information of S_c (unlabeled target domain), where n_1 represents the number of instances in the source domain, n_2 represents the number of instances in the target domain, N_1 is the number of features in the source domain and N_2 is the number of features in the target domain.

Clearly, the core function of HeUDA models is to simultaneously transform S_1 and S_2 to S_c but traditional fuzzy equivalence relations can only do this separately. Thus, the most important work of this paper is to determine how to apply fuzzy equivalence relations to simultaneously transform S_1 and S_2 to S_c. To provide more detail, we propose SFER to let $R_{T\mathcal{D}}^M$s of S_1 and S_2 share the same α, which guarantees that S_1 and S_2 can be clustered as N_c categories with the same α. Figure 2 illustrates the difference between traditional fuzzy equivalence relations and SFER.

Subsection A is the theoretical guarantee for SFER, which is formally demonstrated in subsection B. Subsection C proposes an algorithm to learn the parameters of SFER, and the last subsection introduces how to transfer knowledge from a source domain to a target domain using SFER.

A. Theoretical guarantees

This subsection gives two properties of fuzzy equivalence relations. The first (Theorem 3) demonstrates how many different real numbers exist in the fuzzy equivalence relations matrix and the second (Theorem 4) demonstrates how the number of clusters changes when α changes.

Lemma 2: Given a set $S = \{\mathrm{A}_1, \mathrm{A}_2, \dots, \mathrm{A}_N\}$ and a fuzzy set $\bar{S} = \{\overline{\mathrm{A}_1}, \overline{\mathrm{A}_2}, \dots, \overline{\mathrm{A}_N}\}$ on S, then, for $R_{T\mathcal{D}}^M$ of $\bar{S}$, we have

$$R_{T\mathcal{D}}^M = \begin{pmatrix} R_{(N)}^{N-1} \bigvee \left(R_{(N-2)}^{N-1} \circ r_N \circ r_N^T \circ R_{(N-2)}^{N-1}\right) & R_{(N-1)}^{N-1} \circ r_N \\ r_N^T \circ R_{(N-1)}^{N-1} & 1 \end{pmatrix},$$

where

$$R_{\mathcal{D}}^M = \begin{pmatrix} R^{N-1} & r_N \\ r_N^T & 1 \end{pmatrix}_{N\times N}, R_{(N)}^{N-1} = \underbrace{R^{N-1} \circ \dots \circ R^{N-1}}_{N}, R_{(0)}^{N-1} = \mathbf{I}$$

and R^{N-1} is a $N-1$ by $N-1$ matrix, r_N is a N dimensional vector.

Proof: Based on the max–min operator $\circ$, we arrive at the following equation:

$$R_{\mathcal{D}}^M \circ R_{\mathcal{D}}^M = \begin{pmatrix} R_{(2)}^{N-1} \vee (r_N \circ r_N^T) & R_{(2)}^{N-1} \circ r_N \\ r_N^T \circ R_{(2)}^{N-1} & 1 \end{pmatrix}. \tag{10}$$

In terms of Eq. (10), $R_{T\mathcal{D}}^M = \underbrace{R_{\mathcal{D}}^M \circ R_{\mathcal{D}}^M \circ \dots \circ R_{\mathcal{D}}^M}_{N}$ can be expressed as follows

$$R_{T\mathcal{D}}^M = \begin{pmatrix} R_{(N)}^{N-1} \bigvee_{k=0}^{N-2} \left(R_{(k)}^{N-1} \circ r_N \circ r_N^T \circ R_{(N-2-k)}^{N-1}\right) & R_{(N-1)}^{N-1} \circ r_N \\ r_N^T \circ R_{(N-1)}^{N-1} & 1 \end{pmatrix},$$

where $R_{(k)}^{N-1} \circ r_N \circ r_N^T \circ R_{(N-2-k)}^{N-1}$, $k = 0, \dots, N-2$, cannot satisfy the symmetry but $\bigvee_{k=0}^{N-2}\left(R_{(k)}^{N-1} \circ r_N \circ r_N^T \circ R_{(N-2-k)}^{N-1}\right)$ is a symmetrical matrix. Based on the meaning of operator $\vee$, we have

$$\bigvee_{k=0}^{N-2}\left(R_{(k)}^{N-1} \circ r_N \circ r_N^T \circ R_{(N-2-k)}^{N-1}\right) = R_{(N-2)}^{N-1} \circ r_N \circ r_N^T \circ R_{(N-2)}^{N-1}.$$

So, this lemma is proved. □

Lemma 2 demonstrates the structure of the $R_{T\mathcal{D}}^M$ of $\bar{S}$ from the perspective of the block matrix and provides a useful way to prove Theorem 3.

Theorem 3: Given a set $S = \{\mathrm{A}_1, \mathrm{A}_2, \dots, \mathrm{A}_N\}$ and a fuzzy set $\bar{S} = \{\overline{\mathrm{A}_1}, \overline{\mathrm{A}_2}, \dots, \overline{\mathrm{A}_N}\}$ on S, if the fuzzy relations matrix $R_{\mathcal{D}}^M$ of $\bar{S}$ has $N(N-1)/2+1$ different elements, then $R_{T\mathcal{D}}^M$ of $\bar{S}$ only has N different elements: $r_1 < r_2 < \cdots < r_{N-1} < r_N = 1$.

Proof: We use mathematical induction to prove this theorem based on Lemma 2.

1) First, when N=2, obviously, $R_{T\mathcal{D}}^M$ only has 2 different elements $r_1^2 < r_2^2 = 1$;

2) Then, we assume that the R^{N-1} (the $R_{\mathcal{D}}^M$ of the subset $\overline{S^{N-1}} = \{\overline{\mathrm{A}_1}, \overline{\mathrm{A}_2}, \dots, \overline{\mathrm{A}_{N-1}}\}$) has $(N-1)(N-2)/2$+1 different elements and $R_{(N-2)}^{N-1}$ (the $R_{T\mathcal{D}}^M$ of the subset $\overline{S^{N-1}}$) only has $N-1$ elements: $r_1^{N-1} < r_2^{N-1} < \cdots < r_{N-1}^{N-1} = 1$.

3) Based on Lemma 2, we have

$$R_{T\mathcal{D}}^M = \begin{pmatrix} R_{(N)}^{N-1} \bigvee \left(R_{(N-2)}^{N-1} \circ r_N \circ r_N^T \circ R_{(N-2)}^{N-1}\right) & R_{(N-1)}^{N-1} \circ r_N \\ r_N^T \circ R_{(N-1)}^{N-1} & 1 \end{pmatrix}$$

and we know $R_{(N)}^{N-1}$ only has $N-1$ elements. So, i) we first need to prove $R_{(N)}^{N-1} \vee \left(R_{(N-2)}^{N-1} \circ r_N \circ r_N^T \circ R_{(N-2)}^{N-1}\right)$ only has $N-1$ different elements, then, ii) we need to prove one of the elements in $R_{(N-1)}^{N-1} \circ r_N$ do not exist in $R_{(N)}^{N-1} \vee \left(R_{(N-2)}^{N-1} \circ r_N \circ r_N^T \circ R_{(N-2)}^{N-1}\right)$.

i) We use $r_{ij}^{(N1)}$ to express elements in $R_{(N-2)}^{N-1}$ and $R_{(N)}^{N-1}$ ($R_{(N-2)}^{N-1} = R_{(N)}^{N-1}$), and use $r_{ij}^{(rr)}$ to express elements in $r_N \circ r_N^T$, and use $r_{ij}^{(RrR)}$ to express elements in $R_{(N-2)}^{N-1} \circ r_N \circ r_N^T \circ R_{(N-2)}^{N-1}$. Then, we have

$$r_{ij}^{(RrR)} = \bigvee_{k=1}^{N-1} \bigvee_{k_1=1}^{N-1} r_{ik}^{(N1)} \wedge r_{kk_1}^{(rr)} \wedge r_{k_1 j}^{(N1)}, \tag{11}$$

which means that we need to consider if $r_{ij}^{(RrR)}$ is greater than $r_{ij}^{(N1)}$. Without loss of generality, we select elements (more than 2) equaling r_m^{N-1} ($m < N-1$) as examples: $r_{i_1 j_1}^{(N1)} = r_{i_2 j_1}^{(N1)} = \cdots = r_{i_t j_1}^{(N1)} = r_{i_t j_2}^{(N1)} = \cdots = r_{i_t j_z}^{(N1)} = r_m^{N-1}$ (because $R_{\mathcal{D}}^M$ of $\bar{S}$ has $N(N-1)/2+1$ different elements, there are two of the same elements in the same rows or columns of $R_{T\mathcal{D}}^M$). Based on eq. (11), if one element of $r_{ij}^{(N1)}$ equaling r_m^{N-1} is lower than $r_{ij}^{(RrR)}$, then we have

$$r_{i_1 j_1}^{(RrR)} > r_m^{N-1}, r_{i_2 j_1}^{(RrR)} > r_m^{N-1}, \dots r_{i_t j_1}^{(RrR)} > r_m^{N-1}, \tag{12}$$

$$r_{i_t j_2}^{(RrR)} > r_m^{N-1}, \dots r_{i_t j_z}^{(RrR)} > r_m^{N-1}, \tag{13}$$

meaning that all elements of $r_{ij}^{(N1)}$ equaling r_m^{N-1} will be lower than $r_{ij}^{(RrR)}$ and will be replaced with one element of r_N and elements of $R_{(N-2)}^{N-1}$ in $R_{(N)}^{N-1} \vee \left(R_{(N-2)}^{N-1} \circ r_N \circ r_N^T \circ R_{(N-2)}^{N-1}\right)$. Thus, $R_{(N)}^{N-1} \vee \left(R_{(N-2)}^{N-1} \circ r_N \circ r_N^T \circ R_{(N-2)}^{N-1}\right)$ still has $N-1$ different elements (If there only are two elements equaling

r_m^{N-1}, they only can be replaced with an element of r_N, meaning that $R_{(N)}^{N-1} \vee \left(R_{(N-2)}^{N-1} \circ r_N \circ r_N^T \circ R_{(N-2)}^{N-1}\right)$ still has $N-1$ different elements).

ii) Because $R_{\mathcal{D}}^{M}$ of $\bar{S}$ has $N(N-1)/2+1$ different elements, there is only one maximum element in r_N and this maximum element will only exist in the diagonal of $r_N \circ r_N^T$, denoted by $r_{k_m,k_m}^{(rr)}$. Based on the max-min operator ∘, it is easy to know that the k_m^{th} element of $R_{(N-1)}^{N-1} \circ r_N$ is $r_{k_m,k_m}^{(rr)}$. Then, we prove that $r_{k_m,k_m}^{(rr)}$ does not exist in $R_{(N)}^{N-1} \vee \left(R_{(N-2)}^{N-1} \circ r_N \circ r_N^T \circ R_{(N-2)}^{N-1}\right)$.

If $\exists i, j, s.t. \, r_{ij}^{(RrR)} = r_{k_m,k_m}^{(rr)}$, we have

$$r_{ij}^{(RrR)} = r_{ik_m}^{(N1)} \wedge r_{k_m,k_m}^{(rr)} \wedge r_{k_mj}^{(N1)}, \tag{14}$$

meaning that $r_{ik_m}^{(N1)} \wedge r_{k_mj}^{(N1)} > r_{k_m,k_m}^{(rr)}$. Because

$$r_{ij}^{(N1)} = \bigvee_{k=1}^{N-1} r_{ik}^{(N1)} \wedge r_{kj}^{(N1)} \geq r_{ik_m}^{(N1)} \wedge r_{k_mj}^{(N1)}, \tag{15}$$

we have $r_{ij}^{(N1)} > r_{ij}^{(RrR)} = r_{k_m,k_m}^{(rr)}$, which means that $r_{k_m,k_m}^{(rr)}$ does not exist in $R_{(N)}^{N-1} \vee \left(R_{(N-2)}^{N-1} \circ r_N \circ r_N^T \circ R_{(N-2)}^{N-1}\right)$.

Based on i) and ii), we prove that $R_{T\mathcal{D}}^{M}$ has only N different elements. Combining 1), 2) and 3), this theorem is proved. □

Theorem 3 demonstrates an important property for a fuzzy equivalence matrix and derives Lemma 3 and Theorem 4.

Lemma 3: Given a fuzzy set $\bar{S} = \{\overline{\mathrm{A}_1}, \overline{\mathrm{A}_2}, \dots, \overline{\mathrm{A}_N}\}$, if the fuzzy relations matrix $R_{\mathcal{D}}^{M}$ of $\bar{S}$ has $N(N-1)/2+1$ different elements, then, for $R_{T\mathcal{D}}^{M}$ of $\bar{S}$, we have

$$rank\left(R_{T\mathcal{D}}^{M}(r_i)\right) = rank\left(R_{T\mathcal{D}}^{M}(r_{i-1})\right) + 1, i = 2, \dots, N. \tag{16}$$

Proof: Because $R_{T\mathcal{D}}^{M}(r_{i-1}) \geq R_{T\mathcal{D}}^{M}(r_i)$ holds for $r_i, i = 2, \dots, N$, the element "1" of $R_{T\mathcal{D}}^{M}(r_i)$ will change to "0" when i increasing, meaning that $rank\left(R_{T\mathcal{D}}^{M}(r_i)\right) \geq rank\left(R_{T\mathcal{D}}^{M}(r_{i-1})\right)$ holds. Then, we will prove that $rank\left(R_{T\mathcal{D}}^{M}(r_i)\right)$ is greater than $rank\left(R_{T\mathcal{D}}^{M}(r_{i-1})\right)$.

We use $r_{ls}^{(i)}$ to represent the element in the i^{th} row and s^{th} column of $R_{T\mathcal{D}}^{M}(r_i)$, and $r_{l*}^{(i)}$ to represent elements in i^{th} of $R_{T\mathcal{D}}^{M}(r_i)$. For a specific i, we assume $rank\left(R_{T\mathcal{D}}^{M}(r_{i-1})\right) = k$ and $\forall l_1, l_2 \in I_t^{(i-1)}, r_{l_1*}^{(i-1)} = r_{l_2*}^{(i-1)}$, where $I_t^{(i-1)}$ is a set to collect indicators of same rows of $R_{T\mathcal{D}}^{M}(r_{i-1})$ and $t = 1, 2, \dots, k$. Without loss of generality, we assume that $r_{ls}^{(i-1)} = 1$ but $r_{ls}^{(i)} = 0$ and analyze the value of $rank\left(R_{T\mathcal{D}}^{M}(r_i)\right)$.

Because the reflexivity of $R_{T\mathcal{D}}^{M}(r_{i-1})$, we know $\exists t_0 \; s.t.$

$$\forall l_1, l_2 \in I_{t_0}^{(i-1)}, r_{l_1l_2}^{(i-1)} = 1, \tag{17}$$

and $l \in I_{t_0}^{(i-1)}, s \in I_{t_0}^{(i-1)}$. This means $r_{l*}^{(i)} \neq r_{s*}^{(i)}$ (because $r_{ls}^{(i)} = 0 \neq 1 = r_{ss}^{(i)}$). So, $I_{t_0}^{(i-1)}$ will be divided into two sets when r_{i-1} is replaced with r_i, meaning that $rank\left(R_{T\mathcal{D}}^{M}(r_i)\right) > rank\left(R_{T\mathcal{D}}^{M}(r_{i-1})\right)$.

Based on Theorem 3, we know $rank\left(R_{T\mathcal{D}}^{M}(r_i)\right)$ at most has N values: $rank\left(R_{T\mathcal{D}}^{M}(r_1)\right)$, …, $rank\left(R_{T\mathcal{D}}^{M}(r_N)\right)$. Since $rank\left(R_{T\mathcal{D}}^{M}(r_i)\right) > rank\left(R_{T\mathcal{D}}^{M}(r_{i-1})\right)$ and $rank\left(R_{T\mathcal{D}}^{M}(r_i)\right) \leq N$, we have $rank\left(R_{T\mathcal{D}}^{M}(r_i)\right) = rank\left(R_{T\mathcal{D}}^{M}(r_{i-1})\right) + 1$. □

Theorem 4: Given a fuzzy set $\bar{S} = \{\overline{\mathrm{A}_1}, \overline{\mathrm{A}_2}, \dots, \overline{\mathrm{A}_N}\}$, if the fuzzy relations matrix $R_{\mathcal{D}}^{M}$ of $\bar{S}$ has $N(N-1)/2+1$ different elements and $\alpha \in (r_{i-1}, r_i]$, then $\bar{S}$ will be clustered as i categories, where $\{r_i, i = 1, 2 \dots, N\}$ is the set of N different elements of $R_{T\mathcal{D}}^{M}$ of $\bar{S}$ and $r_1 < r_2 < \cdots < r_{N-1} < r_N = 1$.

Proof: When $\alpha \in (r_{i-1}, r_i]$, $R_{T\mathcal{D}}^{M}(\alpha)$ of $\bar{S}$ is equal to $R_{T\mathcal{D}}^{M}(r_i)$. Based on Lemma 3, we know $rank\left(R_{T\mathcal{D}}^{M}(r_i)\right) = i$, which means that $\bar{S}$ will be clustered as i categories $I_t^{(i)}$, where $I_t^{(i-1)}$ is a set to collect indicators of same rows of $R_{T\mathcal{D}}^{M}(r_{i-1})$ and $\forall t_1, t_2 \in \{1, 2, \dots, i\}, I_{t_1}^{(i)} \cap I_{t_2}^{(i)} = \emptyset$ and $\bigcup I_t^{(i)} = \{1, 2, \dots, N\}$. □

Theorem 4 demonstrates a significant property for fuzzy equivalence: the number of clusters is decided by the value of r_i, which means that we can use r_i of two domains to represent how many clusters both domains have when two domains are applied by the same α.

B. Shared fuzzy equivalence relations (SFER)

Based on subsection A, the aim of the SFER is to let two domains S_1 and S_2 have almost the same $r_i, i = 1, \dots, M = \min(N_1, N_2)$, denoted by r_i^{S1} for S_1 and r_i^{S2} for S_2. Formally, we define a cost function $\mathbf{J_1}(S_1, S_2, \rho_l^{S1}, \rho_k^{S2})$ to express the divergence between r_i^{S1} and r_i^{S2}, which is shown as follows.

$$\mathbf{J_1}(S_1, S_2, \rho_l^{S1}, \rho_k^{S2}) = \sum_{i=1}^{M-1} \left(r_i^{S1}(S_1, \rho_l^{S1}) - r_i^{S2}(S_2, \rho_k^{S2})\right)^2, \tag{18}$$

where ρ_l^{S1} is the parameter vector for elements in S_1 and ρ_k^{S2} is the parameter vector for elements in S_2 (parameters of the triangular membership function). Thus, the SFER aims to minimize the $\mathbf{J}(S_1, S_2, \rho_l^{S1}, \rho_k^{S2})$ by tuning ρ_l^{S1} and ρ_k^{S2}, expressed by

$$\begin{aligned} &\min_{\rho_l^{S1}, \rho_k^{S2}} \mathbf{J_1}(S_1, S_2, \rho_l^{S1}, \rho_k^{S2}) \\ &subject\ to\ \rho_l^{S1} > 0, \\ &\qquad\qquad\quad \rho_k^{S2} > 0. \end{aligned} \tag{19}$$

If the cost function $\mathbf{J_1}(S_1, S_2, \rho_l^{S1}, \rho_k^{S2})$ is approaching 0, r_i^{S1} is almost same as r_i^{S2}, which means that domains S_1 and S_2 will have the same number of clusters when applying the same α to two domains (Fig. 2-(b)).

C. Learning for SFER

To learn the best ρ_l^{S1} and ρ_k^{S2}, we first consider how to minimize $\left(r_{i_0}^{S1} - r_{i_0}^{S2}\right)^2$, where $1 \leq i_0 \leq M-1$ and $r_{i_0}^{S1} < r_{i_0+1}^{S1}$ and $r_{i_0}^{S2} < r_{i_0+1}^{S2}$. Because of the nature of max-min operator, $r_{i_0}^{S1}$ equals one element in $R_{\mathcal{D}}^{M}$ of S_1 and $r_{i_0}^{S2}$ equals one element in $R_{\mathcal{D}}^{M}$ of S_2, which means

$$r_{i_0}^{S1} = exp\left(-\frac{\frac{1}{n} d\left(\mathrm{A}_{l_0}, \mathrm{A}_{l_0'}\right) + \frac{1}{4}\left|\rho_{l_0}^{S1} - \rho_{l_0'}^{S1}\right|}{2\sigma^2}\right), \tag{20}$$

$$r_{i_0}^{S2} = exp\left(-\frac{\frac{1}{n} d\left(\mathrm{B}_{k}, \mathrm{B}_{k_0'}\right) + \frac{1}{4}\left|\rho_{k_0}^{S2} - \rho_{k_0'}^{S2}\right|}{2\sigma^2}\right). \tag{21}$$

It is obvious that $r_{i_0}^{S1}$ is related to $\left|\rho_{l_0}^{S1} - \rho_{l_0'}^{S1}\right|$, so the constrained conditions of $\mathbf{J_1}(S_1, S_2, \rho_l^{S1}, \rho_k^{S2})$ do not affect the value of $\mathbf{J_1}$. Thus, we can define a new optimization problem (no constrained condition) related to ρ_l^{S1} and ρ_k^{S2} as follows:

$$\min_{\rho_l^{S1},\rho_k^{S2}} \mathbf{J_1}(S_1, S_2, \rho_l^{S1}, \rho_k^{S2}). \quad (22)$$

Then, we can obtain the gradients of $\rho_{l_0}^{S1}$, $\rho_{l_0'}^{S1}$, $\rho_{k_0}^{S2}$ and $\rho_{k_0'}^{S2}$ with respect to $\left(r_{i_0}^{S1} - r_{i_0}^{S2}\right)^2$:

$$\frac{\partial\left(r_{i_0}^{S1} - r_{i_0}^{S2}\right)^2}{\partial\rho_{l_0}^{S1}} = \frac{-2\left(r_{i_0}^{S1} - r_{i_0}^{S2}\right)^2 r_{i_0}^{S1}\left|\rho_{l_0}^{S1} - \rho_{l_0'}^{S1}\right|}{2\sigma^2 sign\left(\rho_{l_0}^{S1} - \rho_{l_0'}^{S1}\right)}, \quad (23)$$

$$\frac{\partial\left(r_{i_0}^{S1} - r_{i_0}^{S2}\right)^2}{\partial\rho_{l_0'}^{S1}} = \frac{2\left(r_{i_0}^{S1} - r_{i_0}^{S2}\right)^2 r_{i_0}^{S1}\left|\rho_{l_0}^{S1} - \rho_{l_0'}^{S1}\right|}{2\sigma^2 sign\left(\rho_{l_0}^{S1} - \rho_{l_0'}^{S1}\right)}, \quad (24)$$

$$\frac{\partial\left(r_{i_0}^{S1} - r_{i_0}^{S2}\right)^2}{\partial\rho_{k_0}^{S2}} = \frac{-2\left(r_{i_0}^{S1} - r_{i_0}^{S2}\right)^2 r_{i_0}^{S1}\left|\rho_{k_0}^{S2} - \rho_{k_0'}^{S2}\right|}{2\sigma^2 sign\left(\rho_{k_0}^{S2} - \rho_{k_0'}^{S2}\right)}, \quad (25)$$

$$\frac{\partial\left(r_{i_0}^{S1} - r_{i_0}^{S2}\right)^2}{\partial\rho_{k_0'}^{S2}} = \frac{-2\left(r_{i_0}^{S1} - r_{i_0}^{S2}\right)^2 r_{i_0}^{S1}\left|\rho_{k_0}^{S2} - \rho_{k_0'}^{S2}\right|}{2\sigma^2 sign\left(\rho_{k_0}^{S2} - \rho_{k_0'}^{S2}\right)}. \quad (26)$$

Inspired by incremental gradient descent, for each $r_{i_0}^{S1}$, we can optimize $\rho_{l_0}^{S1}$ and $\rho_{l_0'}^{S1}$ once using the following equations.

$$\rho_{l_0}^{S1} = \rho_{l_0}^{S1} - \eta\frac{\partial\left(r_{i_0}^{S1} - r_{i_0}^{S2}\right)^2}{\partial\rho_{l_0}^{S1}}, \quad (27)$$

$$\rho_{l_0'}^{S1} = \rho_{l_0'}^{S1} - \eta\frac{\partial\left(r_{i_0}^{S1} - r_{i_0}^{S2}\right)^2}{\partial\rho_{l_0'}^{S1}}. \quad (28)$$

Similarly, $\rho_{k_0}^{S2}$ and $\rho_{k_0'}^{S2}$ are optimized once using the following equations.

$$\rho_{k_0}^{S2} = \rho_{k_0}^{S2} - \eta\frac{\partial\left(r_{i_0}^{S1} - r_{i_0}^{S2}\right)^2}{\partial\rho_{k_0}^{S2}}, \quad (29)$$

$$\rho_{l_0'}^{S1} = \rho_{l_0'}^{S1} - \eta\frac{\partial\left(r_{i_0}^{S1} - r_{i_0}^{S2}\right)^2}{\partial\rho_{k_0'}^{S2}}. \quad (30)$$

So, using $r_{i_0}^{S1}$ and $r_{i_0}^{S2}$, we can optimize $\rho_{l_0}^{S1}$, $\rho_{l_0'}^{S1}$, $\rho_{k_0}^{S2}$ and $\rho_{k_0'}^{S2}$ once (no iterations). This means $\rho_{l_0}^{S1}$, $\rho_{l_0'}^{S1}$, $\rho_{k_0}^{S2}$ and $\rho_{k_0'}^{S2}$ can be optimized $M-1$ times using different $r_{i_0}^{S1}$ and $r_{i_0}^{S2}$, where $1 \leq i_0 \leq M-1$. With the optimized $\rho_{l_0}^{S1}$, $\rho_{l_0'}^{S1}$, $\rho_{k_0}^{S2}$ and $\rho_{k_0'}^{S2}$, $R_\mathcal{D}^M$ of S_1 and $R_\mathcal{D}^M$ of S_2 will be updated. Then $\rho_{l_0}^{S1}$, $\rho_{l_0'}^{S1}$, $\rho_{k_0}^{S2}$ and $\rho_{k_0'}^{S2}$ will be optimized M times using different $r_{i_0}^{S1}$ and $r_{i_0}^{S2}$ again. Within limited iterations (or reaching a termination condition), we can obtain the optimized $\rho_{l_0}^{S1}$, $\rho_{l_0'}^{S1}$, $\rho_{k_0}^{S2}$ and $\rho_{k_0'}^{S2}$. Based on these procedures, Algorithm 1 is designed to optimize $\rho_{l_0}^{S1}$, $\rho_{l_0'}^{S1}$, $\rho_{k_0}^{S2}$ and $\rho_{k_0'}^{S2}$ as follows.

```
Algorithm 1. Learning parameters for SFER
    Input: $S_1, S_2$
    Parameter: IterM
1   Randomly generate $\rho_l^{S1}$ and $\rho_k^{S2}$
2   For $i = 1: IterM$
3       Calculate $R_{TD}^M$ of $S_1$ and $S_2$ (based on $\rho_l^{S1}$ and $\rho_k^{S2}$);
4       Extract $r_{i_0}^{S1}$ and $r_{i_0}^{S2}$ from $R_{TD}^M$ of $S_1$ and $S_2$;
5       For $i_0 = 1: M-1$
6           Find $l_0$ and $l_0'$ such that $R_\mathcal{D}^M(l_0, l_0') = r_{i_0}^{S1}$;
7           Find $k_0$ and $k_0'$ such that $R_\mathcal{D}^M(k_0, k_0') = r_{i_0}^{S2}$;
8           Update $\rho_{l_0}^{S1}$, $\rho_{l_0'}^{S1}$, $\rho_{k_0}^{S2}$ and $\rho_{k_0'}^{S2}$ using Eqs. (27)-(30);
9       end
10  end
11  $\rho_l^{S1} = \rho_l^{S1} + \min(\rho_l^{S1}) + \epsilon$;
12  $\rho_k^{S2} = \rho_k^{S2} + \min \rho_k^{S2} + \epsilon$.
```

In Algorithm 1, lines 11 and 12 ensure that ρ_l^{S1} and ρ_k^{S2} are more than 0 ($\epsilon > 0$).

D. HEUDA via SFER (F-HeUDA)

In this section, we introduce how to select α, how to generate S_C and how to transfer knowledge from the source domain to the target domain.

After executing Algorithm 1, we obtain $r_{i_0}^{S1}$ and $r_{i_0}^{S2}$ generated by the best ρ_l^{S1} and ρ_k^{S2}, $i_0 = 0, \dots, M-1$. So, the intervals of sharing α between two domains can be calculated (two domains can share the same α when α is in these intervals). These intervals can be expressed as follows:

$$\left[\max\left(r_{i_0}^{S1}, r_{i_0}^{S2}\right), \min\left(r_{i_0+1}^{S1}, r_{i_0+1}^{S2}\right)\right), \quad (31)$$

where $r_0^{S1} = r_0^{S2} = 0$ and $r_M^{S1} = r_M^{S2} = 1$. We obtain following intervals, $[0, \min(r_1^{S1}, r_1^{S2}))$, $[\max(r_1^{S1}, r_1^{S2}), \min(r_2^{S1}, r_2^{S2}))$, …, $[\max(r_{M-1}^{S1}, r_{M-1}^{S2}), 1)$. If the α, in these intervals, is selected, S_1 and S_2 have the same number of clusters. Then, we select the α which is in the largest interval as the best α (because two domains can share most information in this largest interval).

Remark 2: *If we select the* $\alpha \in \left[r_{i_0}^{S1}, r_{i_0+1}^{S1}\right)$, *we do know* S_1 *is clustered into* $i_0 + 1$ *clusters but do not guarantee that* S_2 *also*

TABLE I
DESCRIPTIONS OF THE FOUR DATASETS

Field	Data Name	# of Instances	# of Attributes
Credit Assessment (two datasets)	German Credit Data	1000	24
	Australian Credit Approval	690	14
Cancer Detection (two datasets)	Breast Cancer Wisconsin (Original) Dataset	699	9
	Breast Cancer Wisconsin (Diagnostic) Dataset	569	30

TABLE II
TRANSFER TASKS IN TWO FIELDS (4 TASKS IN TOTAL)

Field	Source Domain	Target Domain	Labels	Task Name
Credit Assessment (two datasets)	German Credit Data	Australian Credit Approval	1: Good	G2A
	Australian Credit Approval	German Credit Data	1: Good	A2G
Cancer Detection (two datasets)	Breast Cancer Wisconsin (Original) dataset	Breast Cancer Wisconsin (Diagnostic) dataset	1: Malignant	CO2CD
	Breast Cancer Wisconsin (Diagnostic) dataset	Breast Cancer Wisconsin (Original) dataset	1: Malignant	CD2CO

has $i_0 + 1$ clusters. Thus, we need to select α belonging to $\left[max\left(r_{i_0}^{S1}, r_{i_0}^{S2}\right), min\left(r_{i_0+1}^{S1}, r_{i_0+1}^{S2}\right)\right)$ to make sure S_1 and S_2 have $i_0 + 1$ clusters.

Based on the SFER (with the best α), both $S_1 = \{\mathrm{A}_1, \mathrm{A}_2, \ldots, \mathrm{A}_{N_1}\}$ and $S_2 = \{\mathrm{B}_1, \mathrm{B}_2, \ldots, \mathrm{B}_{N_2}\}$ can be clustered as N_c clusters, such as

$$S_1 = \left\{\{\mathrm{A}_1, \mathrm{A}_2\}_1, \{\mathrm{A}_3, \mathrm{A}_4, \mathrm{A}_5\}_2, \ldots, \{\mathrm{A}_{N_1}\}_{N_c}\right\}, \qquad (32)$$

$$S_2 = \left\{\{\mathrm{B}_1\}_1, \{\mathrm{B}_2, \mathrm{B}_3, \mathrm{B}_4\}_2, \ldots, \{\mathrm{B}_{N_1-1}, \mathrm{B}_{N_1}\}_{N_c}\right\}. \qquad (33)$$

Next, we generate the latent features of the two domains using an addition operator, such as

$$S_1^{latent} = \{\mathrm{A}_1 + \mathrm{A}_2, \mathrm{A}_3 + \mathrm{A}_4 + \mathrm{A}_5, \ldots, \mathrm{A}_{N_1}\}, \qquad (34)$$

$$S_2^{latent} = \{\mathrm{B}_1, \mathrm{B}_2 + \mathrm{B}_3 + \mathrm{B}_4, \ldots, \mathrm{B}_{N_1-1} + \mathrm{B}_{N_1}\}. \qquad (35)$$

Then, the standard score is used to normalize each latent feature and we have

$$S_1^{common} = \{f(\mathrm{A}_1 + \mathrm{A}_2), f(\mathrm{A}_3 + \mathrm{A}_4 + \mathrm{A}_5), \ldots, f(\mathrm{A}_{N_1})\}, \quad (36)$$

$$S_2^{common} = \{f(\mathrm{B}_1), f(\mathrm{B}_2 + \mathrm{B}_3 + \mathrm{B}_4), \ldots, f(\mathrm{B}_{N_1-1} + \mathrm{B}_{N_1})\}. (37)$$

where $f(\cdot)$ represents the standard score function. Thus, we obtain the common feature set $S_c = \{\mathrm{C}_1, \mathrm{C}_2, \ldots, \mathrm{C}_{N_c}\}$, $\mathrm{C}_i \in \mathbb{R}^{n_c}$, $n_c = n_1 + n_2$ (without loss of generality, the former n_1 instances in S_c comes from source domain). Finally, we can use the labeled information of S_c (knowledge from the source domain) to label the unlabeled information of S_c (unlabeled target domain) using a support vector machine (SVM).

V. Experiments

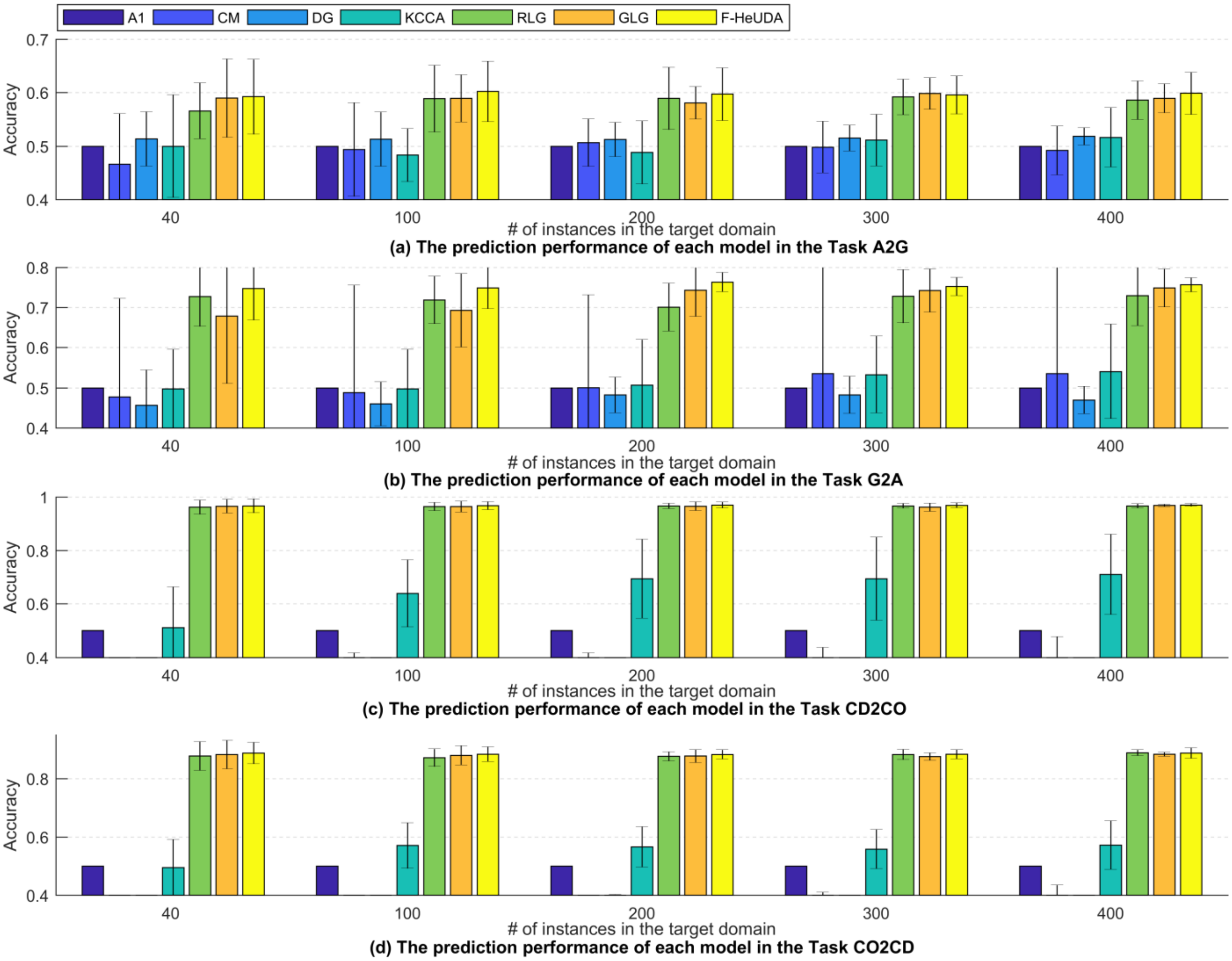
In this section, we use four real datasets to test the proposed F-HeUDA model and analyze the convergence of Algorithm 1 and how α influences the model's performance.

A. Dataset description and parameters setting

To validate the effectiveness of the proposed model on small datasets, we select four datasets from the UCI Machine Learning Repository (http://archive.ics.uci.edu/ml/index.html). The details of these datasets are provided in Table I. For each dataset, we generate four transfer tasks, as shown in Table II. Each task is described in detail as follows:

1) Task 1- G2A: Assume that the German data is labeled and the Australian data is unlabeled and has far fewer instances than the German data. Label "1" means "good credit" and label "2" means "bad credit". This task aims to answer the question: "Can we use knowledge from German credit records to label unlabeled Australian data (small dataset)?"

2) Task 2-A2G: Assume that the Australian data is labeled and the German data is unlabeled and has far fewer instances than the Australian data. Label "1" means "good credit" and label "2" means "bad credit". This task aims to answer the question: "Can we use knowledge from Australian credit



(a) The prediction performance of each model in the Task A2G

(b) The prediction performance of each model in the Task G2A

(c) The prediction performance of each model in the Task CD2CO

(d) The prediction performance of each model in the Task CO2CD

Fig. 3. The performance of each model (mean accuracy and standard deviation) on 4 tasks. In each subfigure, the minimum mean accuracy is set as 0.4.

TABLE III
THE OVERALL PERFORMANCE OF EACH MODEL ON FOUR TASKS

Tasks	A1	CM	DG	KCCA	RLG	GLG	F-HeUDA
A2G	50.00%	49.15%	51.48%	50.01%	58.46%	58.98%	***59.74%***
G2A	50.00%	50.75%	47.02%	51.52%	72.10%	72.12%	***75.38%***
CD2CO	50.00%	22.04%	19.54%	65.00%	96.59%	96.57%	***96.93%***
CO2CD	50.00%	23.40%	26.96%	55.24%	87.94%	87.98%	***88.51%***

TABLE IV
THE OVERALL STANDARD DEVIATION OF EACH MODEL ON FOUR TASKS

Tasks	A1	CM	DG	KCCA	RLG	GLG	F-HeUDA
A2G	-	6.43%	***3.49%***	6.17%	4.83%	4.07%	4.99%
G2A	-	25.78%	5.37%	10.45%	6.70%	8.52%	***3.90%***
CD2CO	-	20.29%	12.12%	14.65%	1.37%	1.65%	***1.33%***
CO2CD	-	15.76%	8.65%	7.88%	2.45%	2.46%	***2.23%***

records to label unlabeled German data (small dataset)?”

3) Task 3-CO2CD: Assume that in the Breast Cancer Wisconsin (Original) dataset (CO in Table II) “1” represents “malignant” and “0” represents “benign”. Another unlabeled dataset related to breast cancer also exists but has far fewer instances than the CO dataset. This task aims to answer the question: “Can we use the knowledge from CO to label ‘malignant’ in the unlabeled small dataset?”

4) Task 4-CD2CO: Assume that in the Breast Cancer Wisconsin (Diagnostic) dataset (CD in Table II) “1” represents “malignant” and “0” represents “benign”. Another unlabeled dataset related to breast cancer also exists. This task aims to answer the question: “Can we use the knowledge from CD to label ‘malignant’ in the small unlabeled dataset?”

For the Algorithm 1, this paper sets IterM as 1000, σ as 3 and η as 0.5. For SVM, we adopt LIBSVM with default parameters: the SVM type is C-SVM with C equaling to 1, kernel function is radial basis function with gamma equaling to $\frac{1}{\#\ of\ Attributes}$, epsilon, the tolerance of termination criterion, is set as 0.001.

B. Prediction performance

In this section, we describe the prediction performance of the proposed model and benchmarks. We select two non-transfer models, A1 and CM, as baselines and three transfer models, dimension reduction GFK (DG), KCCA [26], random linear monotonic maps GFK (RLG) [27], GLG [27] as the main benchmarks. The A1 model labels all instances of a target domain with 1 and the CM model clusters instances of a target domain and gives each instance a pseudo label (clustering model is the fuzzy c-means model with). The DG model is based on dimension reduction technology, introduced in [27] as a benchmark of HeUDA models, and KCCA is based on canonical correlation analysis and requires both domains to have the same number of instances. The RLG and GLG models are proposed in our previous work. The former does not require both domains to have the same number of instances but the latter requires this condition. To test these models’ prediction performance when a target domain is a small dataset, we carried out experiments when n_2 (the number of instances in a target domain) is 40, 100, 200, 300, 400. For each model, we carried out the experiments 20 times to obtain reliable results.

The results, as illustrated in Fig. 3, clearly show the prediction performance of each model. From this figure, it is apparent that the proposed model outperforms the others when n_2 is small. The CM and DG models have lower mean accuracy and higher standard deviation than the other models in most cases. For the KCCA model, its performance increases when n_2 increases but its accuracy is always lower than RLG, GLG and F-HeUDA. This is because the KCCA model cannot reliably transfer knowledge from the source domain to the target domain. The RLG model has better performance than the GLG model when n_2 is 40 and 100 in some cases, but it is still unstable when n_2 increases. The GLG model has good performance when n_2 increases but it cannot guarantee reliable results when n_2 is small (see task G2A). From Fig. 3, the following results can be observed.

1) The KCCA model has a worse performance than non-transfer models in some cases;

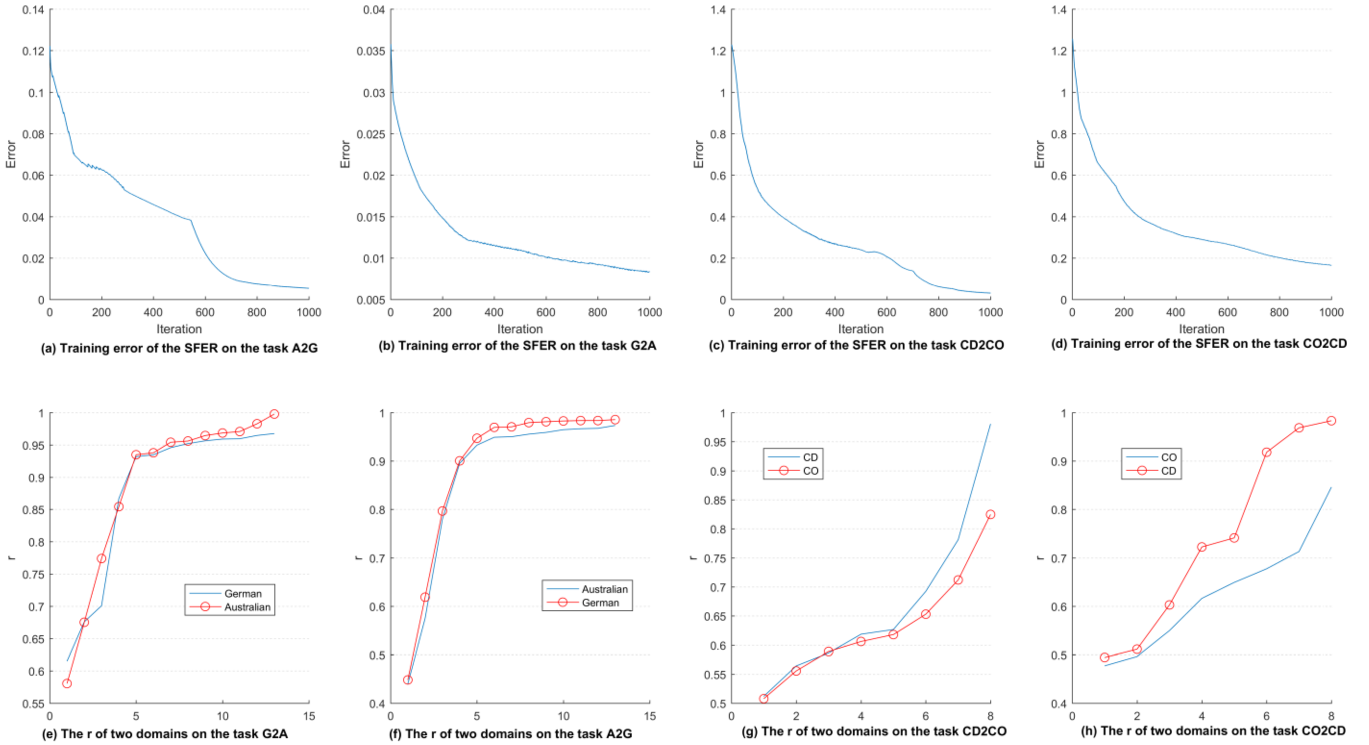
2) The RLG, GLG and F-HeUDA models have more stable results than the non-transfer models and DG and KCCA;

3) Compared to RLG and GLG, the proposed model has a better performance when the number of instances in the target domain is small (<200);

4) Although the GLG model is worse than the proposed model when n_2 is small, the performance of the GLG model improves when n_2 increases;

5) When n_2 increases, the performance of the proposed model approaches the performance of the GLG model.

Table III shows the overall prediction performance of each model (averaging the mean accuracy of each model when n_2 is 40, 100, 200, 300, 400). It is apparent that the proposed model is better than the benchmarks. The proposed model needs much less running time than the GLG model: the F-HeUDA takes 16 seconds to label 400 instances of the target domain while the



(a) Training error of the SFER on the task A2G

(b) Training error of the SFER on the task G2A

(c) Training error of the SFER on the task CD2CO

(d) Training error of the SFER on the task CO2CD

(e) The r of two domains on the task G2A

(f) The r of two domains on the task A2G

(g) The r of two domains on the task CD2CO

(h) The r of two domains on the task CO2CD

Fig. 4. The convergence of Algorithm 1 on four tasks. Subfigures (a)-(d) illustrate the value of the cost function $\mathbf{J_1}$ and subfigures (e)-(f) illustrate the r_i of R_{TD}^{M} of the source domain and the target domain.

TABLE V
THE PREDICTION PERFORMANCE OF F-HEUDA FOR α-CUT DECISION MAKING

Tasks	$\alpha = 0.1$	$\alpha = 0.3$	$\alpha = 0.5$	$\alpha = 0.7$	$\alpha = 0.9$
A2G	59.90%±4.19%	59.05%±3.73%	***62.35%±4.53%***	60.85%±4.67%	58.30%±5.99%
G2A	74.20%±5.55%	***75.80%±4.84%***	75.65%±4.17%	73.55%±9.17%	56.75%±15.33%
CD2CO	96.60%±1.50%	***97.40%±1.31%***	96.85%±1.50%	97.00%±1.41%	95.00%±2.70%
CO2CD	89.50%±2.98%	88.70%±2.68%	91.40%±2.56%	***92.15%±3.83%***	90.75%±3.77%

GLG model needs 107 seconds to finish the same work with the proposed model, which means that the proposed model is still a potential choice even when n_2 is a large number. Except for accuracy, this section also discusses the stability of each model using standard deviation. The mean overall standard deviation of each model is listed in Table IV. From Table III and Table IV, we can obtain following results.

1) The F-HeUDA model is the most stable model for three tasks: G2A, CD2CO and CO2CD;

2) For tasks CD2CO and CO2CD, RLG, GLG and F-HeUDA models have much better performance in terms of stability than the other models;

3) The KCCA model is the most unstable model among DG, KCCA, RLG, GLG and F-HeUDA, which means that the KCCA model experiences difficulty in correctly transferring knowledge from the source domain to the target domain;

4) When n_2 is small, the GLG model becomes unstable, mainly because the GLG model only uses a few instances of the source domain (the transferable knowledge of the GLG model is limited by n_2);

5) Although the RLG model uses all instances of the source domain, its random map nature limits its ability to obtain good feature representation.

C. *Convergence of learning algorithm*

This section discusses the convergence of the learning algorithm for the parameters of the SFER. Figure 4 illustrates the convergence of Algorithm 1 on four tasks when the target domain has 400 instances. It is apparent that the proposed algorithm can effectively optimize the parameters of SFER. From subfigures Fig. 4-(e)-(h), we can see that the two domains have almost the same r_i after optimizing SFER, which means that two domains can share almost every $\alpha \in [0, 1]$ (like Fig. 2-(b)). For a different task, Algorithm 1 has a different performance. For example, for 1000 iterations, the G2A and A2G tasks have fewer errors than the CD2CO and CO2CD tasks, mainly because the divergence of the number of features of the CD and CO datasets is greater than that of the German and Australian datasets. This indicates that for a high divergence task (divergence of the number of features of two domains), the IterM should be set as a larger number to ensure the performance of Algorithm 1.

D. "$\alpha - cut$" decision making

When Algorithm 1 has optimized the parameters of the SFER, two domains can apply the same α to obtain the same number of clusters, which provides a way to transfer knowledge from the source domain to the target domain. In section IV-D, we propose to adopt the α which is in the largest interval as the best α because two domains can share the most information in the largest interval. However, the selection of α is actually a decision-making issue (different users may select different α based on their requirements), so it is necessary to discuss how the selection of α influences the performance, which provides another way to analyze the SFER model. In this section, we let the number of instances of the target domain be 100 and test how the prediction performance changes when α is set as 0.1, 0.3, 0.5, 0.7 and 0.9. Similarly, we demonstrate the results on four tasks and each experiment is carried out 20 times.

Table V gives the detailed prediction performance of F-HeUDA when α changes on each task. First, we analyze the relationship between the α selected by the F-HeUDA and the best α shown in Table V. 1) For task A2G, the α selected by the F-HeUDA is around *0.6112* (after running the experiment 20 times) and Table V shows that the best α is around 0.5. 2) For task G2A, the F-HeUDA selects α as *0.2674* and Table V shows that the best α is around 0.3. 3) For task CD2CO, the α selected by the F-HeUDA is around *0.3729* and the best α shown in Table V is around 0.3. 4) For task CO2CD, the F-HeUDA selected *0.4689* as the α and Table V shows that the best α is around 0.7. Based on these results and Table V, we can conclude the following:

1) Except for task CO2CD, the α selected by the F-HeUDA is near the best α as shown in Table V;

2) When α lies in the interval [0.3 0.7], the F-HeUDA model obtains a better performance;

3) When α increases, the standard deviation of the proposed model is higher, especially for task G2A.

These results demonstrate that the best α will not be near to extreme numbers (such as 0 or 1), which is consistent with normal decision-making scenarios (extreme decisions rarely occur). So, the F-HeUDA is a suitable model for decision making and its method of automatically selecting α is effective and accurate.

VI. CONCLUSION AND FUTURE STUDIES

In this paper, we applied fuzzy equivalence relations into heterogeneous unsupervised domain adaptation, the most challenging issue in the field, through developing the F-HeUDA model. We first propose a metric $\mathcal{D}$ on an n-dimensional fuzzy space $\mathcal{F}(\mathbb{R}^n)$ and use this metric to measure the similarity between two fuzzy vectors and to build the fuzzy equivalence relations matrixes of the source domain and the target domain. Then, based on two discovered properties of the fuzzy equivalence relations, we propose the SFER model which lets two domains share the same α and obtain the same number of clustering categories. Eventually, through these clustering categories, knowledge from the source domain is successfully transferred to the target domain where two domains have a different number of instances. This paper shows the potentiality of fuzzy technologies to address the HeUDA issues and outlines a new way (like the SFER) to obtain common representations of two different domains using fuzzy technologies.

Compared to existing HeUDA models, the proposed F-HeUDA model overcomes the drawbacks of CCA and the Grassmann manifold (both need two domains that have the same number of instances) and works well when the target domain has very few instances. It means that the proposed model is suitable to address the issue of small datasets via transferring knowledge from big datasets to small datasets. To validate the performance of the proposed model, we selected four real datasets to generate four transfer tasks. The prediction accuracy and stability of the proposed model are better than those of the benchmarks. These results show the superiority of the proposed model lies in transferring more knowledge from the source domain to the target domain. From the theoretical perspective, this paper proves that $\left(\mathcal{D}, \mathcal{F}(\mathbb{R}^n)\right)$ is a metric space and proves two important properties of the fuzzy equivalence relations. Both properties are key to the performance of the SFER model.

This study particularly presents how fuzzy techniques can be applied in transfer learning to produce advanced learning models. Our future work includes: 1) the automatic selection of source domains for a specific target domain by applying the metric on the fuzzy geometry; 2) the extension of fuzzy equivalence relations based SFER model for multiple domains; and 3) development of new F-HeUDA via multiple source domains.

ACKNOWLEDGEMENT

The work presented in this paper was supported by the Australian Research Council (ARC) under Discovery Grant DP170101632.

REFERENCES

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

[2] J. Lu, V. Behbood, P. Hao, H. Zuo, S. Xue, and G. Zhang, "Transfer learning using computational intelligence: A survey," *Knowledge-Based Syst.*, vol. 80, pp. 14–23, 2015.

[3] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE Trans. Neural Networks*, vol. 22, no. 2, pp. 199–210, 2011.

[4] B. Gong, K. Grauman, and F. Sha, "Learning kernels for unsupervised domain adaptation with applications to visual object recognition," *Int. J. Comput. Vis.*, vol. 109, no. 1–2, pp. 3–27, 2014.

[5] 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, 2017.

[6] M. Xiao and Y. Guo, "Feature space independent semi-supervised domain adaptation via kernel matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 1, pp. 54–66, 2015.

[7] Y. Fu, T. M. Hospedales, T. Xiang, and S. Gong, "Transductive multi-view zero-shot learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 11, pp. 2332–2345, 2015.

[8] F. Liu, X. Xu, S. Qiu, C. Qing, and D. Tao, "Simple to Complex Transfer Learning for Action Recognition," *IEEE Trans. Image Process.*, vol. 25, no. 2, pp. 949–960, 2016.

[9] 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, no. 7, pp. 1414–1430, 2017.

[10] N. Courty, R. Flamary, D. Tuia, S. Member, and A. Rakotomamonjy, "Optimal transport for domain adaptation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 9, pp. 1853–1865, 2017.

[11] 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. Accept, 2017.

[12] Y. Aytar and A. Zisserman, "Tabula Rasa : Model Transfer for Object Category Detection," in *Proceedings of the 13th International Conference on Computer Vision*, 2011, pp. 2252–2259.

[13] J. Yang, R. Yan, and A. G. Hauptmann, "Cross-domain video concept detection using adaptive SVMs," in *Proceedings of the 15th ACM International Conference on Multimedia*, 2007, pp. 188–197.

[14] A. Bergamo and L. Torresani, "Exploiting weakly-labeled web images to improve object classification : A domain adaptation approach," in *Proceedings of the 24th Annual Conference on Neural Information Processing Systems*, 2010, pp. 181–189.

[15] J. Hoffman, U. C. B. Eecs, E. Rodner, U. C. B. Eecs, T. Darrell, U. C. B. Eecs, J. Donahue, U. C. B. Eecs, and K. Saenko, "Efficient learning of domain-invariant image representations," in *Proceedings of the 1st International Conference on Learning Representations*, 2013, pp. 1–9.

[16] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2066–2073.

[17] Y. Shi, L. Angeles, and F. Sha, "Information-theoretical learning of discriminative clusters for unsupervised domain adaptation," in *Proceedings of the 29th International Conference on Machine Learning*, 2012, pp. 1079–1086.

[18] M. Long, J. Wang, Y. Cao, J. Sun, and P. S. Yu, "Deep learning of transferable representation for scalable domain adaptation," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 8, pp. 2027–2040, 2016.

[19] A. Gretton, "A kernel two-sample test," *J. Mach. Learn. Res.*, vol. 13, pp. 723–773, 2012.

[20] R. Gopalan, R. Li, and R. Chellappa, "Unsupervised adaptation across domain shifts by generating intermediate data representations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 11, pp. 2288–2302, 2014.

[21] W. Li, L. Duan, D. Xu, and I. W. Tsang, "Learning with augmented features for supervised and semi-supervised heterogeneous domain adaptation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 6, pp. 1134–1148, 2014.

[22] B. Kulis, K. Saenko, and T. Darrell, "What you saw is not what you get: Domain adaptation using asymmetric kernel transforms," in *Proceedings of the 24th IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2011, pp. 1785–1792.

[23] X. Shi, Q. Liu, W. Fan, and P. S. Yu, "Transfer across completely different feature spaces via spectral embedding," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 4, pp. 906–918, 2013.

[24] C. Wang and S. Mahadevan, "Heterogeneous domain adaptation using manifold alignment," in *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, 2011, pp. 1541–1546.

[25] H. V Nguyen, H. T. Ho, S. Member, and V. M. Patel, "DASH-N : Joint hierarchical domain adaptation and feature learning," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5479–5491, 2015.

[26] Y. R. Yeh, C. H. Huang, and Y. C. F. Wang, "Heterogeneous domain adaptation and classification by exploiting the correlation subspace," *IEEE Trans. Image Process.*, vol. 23, no. 5, pp. 2009–2018, 2014.

[27] F. Liu, G. Zhang, and J. Lu, "Heterogeneous transfer learning: An unsupervised approach," *arXiv:1701.02511 [cs.LG]*, pp. 1–48, 2017.

[28] 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, 2016.

[29] V. Behbood, S. Member, J. Lu, and G. Zhang, "Fuzzy Refinement Domain Adaptation for Long Term Prediction in Banking Ecosystem," *IEEE Trans. Fuzzy Syst.*, vol. 10, no. 2, pp. 1637–1646, 2014.

[30] V. Behbood, J. Lu, G. Zhang, and W. Pedrycz, "Multistep fuzzy bridged refinement domain adaptation algorithm and its application to bank failure prediction," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 6, pp. 1917–1935, 2015.

[31] Z. Deng, Y. Jiang, F. L. Chung, H. Ishibuchi, and S. Wang, "Knowledge-leverage-based fuzzy system and its modeling," *IEEE Trans. Fuzzy Syst.*, vol. 21, no. 4, pp. 597–609, 2013.

[32] Z. Deng, Y. Jiang, K. S. Choi, F. L. Chung, and S. Wang, "Knowledge-leverage-based TSK fuzzy system modeling," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 24, no. 8, pp. 1200–1212, 2013.

[33] Z. Deng, Y. Jiang, H. Ishibuchi, K.-S. Choi, and S. Wang, "Enhanced Knowledge-Leverage-Based TSK Fuzzy System Modeling for Inductive Transfer Learning," *ACM Trans. Intell. Syst. Technol.*, vol. 8, no. 1, pp. 1–21, 2016.

[34] Z. Deng, I. S. Member, Y. Jiang, I. Member, and F. Chung, "Transfer Prototype-based Fuzzy Clustering," *Trans. Fuzzy Syst.*, vol. 24, no. 5, pp. 1210–1232, 2016.

[35] Z. Deng, K. S. 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, 2014.

[36] S. Sun, J. Yun, H. Lin, N. Zhang, A. Abraham, and H. Liu, "Granular transfer learning using type-2 fuzzy HMM for text sequence recognition," *Neurocomputing*, vol. 214, pp. 126–133, 2016.

[37] D. Ghosh and D. Chakraborty, "Analytical fuzzy plane geometry I," *Fuzzy Sets Syst.*, vol. 209, pp. 66–83, 2012.

[38] D. Chakraborty and D. Ghosh, "Analytical fuzzy plane geometry II," *Fuzzy Sets Syst.*, vol. 243, pp. 84–109, 2014.

[39] D. Ghosh and D. Chakraborty, "Analytical fuzzy plane geometry III," *Fuzzy Sets Syst.*, vol. 283, pp. 83–107, 2016.

[40] J. J. Buckley and E. Eslami, "Fuzzy plane geometry I : Points and lines," *Fuzzy Sets Syst.*, vol. 86, pp. 179–187, 1997.

[41] J. J. Buckley and E. Eslami, "Fuzzy plane geometry II : Circles and polygons," *Fuzzy Sets Syst.*, vol. 87, pp. 79–85, 1997.

[42] Y. Li, Q. Huang, W. Xie, and X. Li, "A novel visual codebook model based on fuzzy geometry for large-scale image classification," *Pattern Recognit.*, vol. 48, no. 10, pp. 3125–3134, 2015.

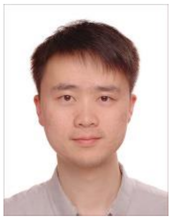
[43] R. Goetschel and W. Voxman, "Topological properties of fuzzy numbers," *Fuzzy Sets Syst.*, vol. 10, pp. 87–99, 1983.

[44] L. A. Zadeh, "Similarity relations and fuzzy orderings," *Inf. Sci.*, vol. 3, pp. 177–200, 1971.

[45] A. Mirzaei and M. Rahmati, "A novel hierarchical-clustering-combination scheme based on fuzzy-similarity relations," *IEEE Trans. Fuzzy Syst.*, vol. 18, no. 1, pp. 27–39, 2010.

[46] G. J. Klir and B. Y. Yklsit, *Fuzzy sets and fuzzy logic: theory and applications*, 1st ed. Prentice Hall PTR, 1995.

[47] O. Kaleva and S. Seikkala, "On fuzzy metric spaces," *Fuzzy Sets Syst.*, vol. 12, pp. 215–229, 1984.



Feng Liu received a M.S. in probability and statistics and a B.S. in mathematics from the School of Mathematics and Statistics, Lanzhou University, China, in 2015 and 2013, respectively. He is working toward a Ph.D. with the Faculty of Engineering and Information Technology, University of Technology Sydney, Australia. His research interests include transfer learning and domain adaptation. He is a Member of the Decision Systems and e-Service Intelligence (DeSI) Research Laboratory, Center for Artificial Intelligence, University of Technology Sydney.

Jie Lu (F'18) is a Distinguished Professor and the Director of Centre for Artificial Intelligence at the University of Technology Sydney, Australia. She received the Ph.D. degree from Curtin University of Technology, Australia, in 2000.

Her main research expertise is in fuzzy transfer learning, decision support systems, concept drift, and recommender systems. She has published six research books and 400 papers in Artificial Intelligence, IEEE transactions on Fuzzy Systems and other refereed journals and conference proceedings. She has won over 20 Australian Research Council (ARC) discovery grants and other research grants for over $4 million. She serves as Editor-In-Chief for Knowledge-Based Systems (Elsevier) and Editor-In-Chief for International Journal on Computational Intelligence Systems (Atlantis), has delivered 20 keynote speeches at international conferences, and has chaired 10 international conferences. She is a Fellow of IEEE and Fellow of IFSA.

Guangquan Zhang is an Associate Professor and Director of the Decision Systems and e-Service Intelligent (DeSI) Research Laboratory in Centre for Artificial Intelligence, Faculty of Engineering and Information Technology, University of Technology Sydney, Australia. He received a Ph.D. in applied mathematics from Curtin University of Technology, Australia, in 2001.

His research interests include fuzzy machine learning, fuzzy optimization, and machine learning and data analytics. He has authored four monographs, five textbooks, and 350 papers including 160 refereed international journal papers.

Dr Zhang has been awarded seven Australian Research Council (ARC) Discovery Project grants and many other research grants. He was awarded an ARC QEII fellowship in 2005. He has served as a member of the editorial boards of several international journals, as a guest editor of eight special issues for IEEE transactions and other international journals, and co-chaired several international conferences and workshops in the area of fuzzy decision-making and knowledge engineering.