Enhanced Knowledge-Leverage Based TSK Fuzzy System Modeling for Inductive Transfer Learning

ZHAOHONG DENG, School of Digital Media, Jiangnan University
YIZHANG JIANG, School of Digital Media, Jiangnan University
HISAO ISHIBUCHI, Osaka Prefecture University
KUP-SZE CHOI, The Hong Kong Polytechnic University
SHITONG WANG, School of Digital Media, Jiangnan University

The knowledge-leverage based Takagi–Sugeno–Kang fuzzy system (KL-TSK-FS) modeling method has shown the promising performance for fuzzy modeling tasks where transfer learning is required. However, the knowledge-leverage mechanism of the KL-TSK-FS can be further improved. This is because available training data in a target domain are not utilized for the learning of antecedents and the knowledge transfer mechanism from a source domain to the target domain is still too simple for the learning of consequents when a Takagi–Sugeno–Kang fuzzy system (TSK FS) mode is trained in the target domain. The proposed method, i.e. the enhanced KL-TSK-FS (EKL-TSK-FS), has two knowledge-leverage strategies for enhancing the parameter learning of the TSK FS model for the target domain using available information from the source domain. One strategy is used for the learning of antecedent parameters while the other is for consequent parameters. It is demonstrated that the proposed EKL-TSK-FS has higher transfer learning abilities than the KL-TSK-FS. In addition, the EKL-TSK-FS has been further extended for the multi-source scene.

• Enhanced KL-TSK-FS, Fuzzy systems • Knowledge leverage • Missing data • Fuzzy modeling and Transfer learning.

ACM Reference Format:

Zhaohong Deng, Yizhang Jiang, Hisao Ishibuchi, Kup-sze Choi, and Shitong Wang, 2015. Enhanced Knowledge-Leverage Based TSK Fuzzy System Modeling for Inductive Transfer Learning. *ACM Trans. Intelligent Systems and Technology*.

1. INTRODUCTION

Most modeling methods require sufficient data to be collected for model learning. On the one hand, in many real-world applications the available data may be insufficient since the data is scarce or very noisy. In this situation, many traditional modeling methods become unfeasible. On the other hand, for a current scene, there are usually some reference scenes along with a great deal of useful information. While the current and reference scenes are not the same, they are similar to each other. An interesting topic for research is how to utilize the available information from the reference scenes in the modeling for the current scene. Transfer learning is the technique used to address this topic [1-4, 49-51], where the current scene and the reference scene are usually called the target domain and the source domain, respectively.

In Fig. 1, an illustration of transfer learning is given. For clarity, several terms relating to transfer learning used in this paper are described here [5, 6]. (1) *Domain*: A domain is a scene where a modeling task is to be accomplished. It is usually characterized by the data collected in this domain and the learning task to be performed in this domain. (2) *Target domain*: In transfer learning, a current scene is

This work was supported in part by the Jiangsu Province Outstanding Youth Fund under grant BK20140001, the National Natural Science Foundation of China under grants 61170122 and 61272210, and the Ministry of Education's Program for New Century Excellent Talents under grant NCET-120882.

Authors' affiliations: Z. Deng, Y. Jiang, and S. Wang, School of Digital Media, Jiangnan University, P.R. China; H. Ishibchi, Department of Computer Science and Intelligent Systems, Osaka Prefecture University, Japan; K. Choi, Centre for Smart Health, The Hong Kong Polytechnic University, Hong Kong.

called a target domain, where a modeling task is to be performed. It is usually assumed that the target domain does not have enough data or information for proper modeling. (3) *Source domain*: A reference scene is usually called a source domain, which has some similarities to the target domain in data distributions and/or learning tasks. While the source and target domains are different, it is assumed in transfer learning that the source domain can provide useful information for the modeling task of the target domain.

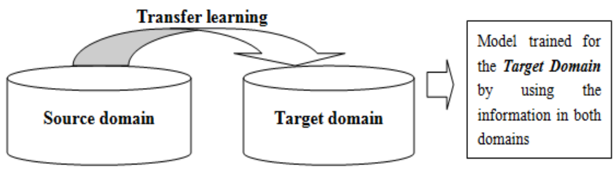

Fig. 1 An illustration of transfer learning

Transfer learning has recently been studied extensively for different learning tasks, including supervised learning [1, 3, 5-19, 41-46, 48] and unsupervised learning [4, 20-22, 47]. Various transfer learning methods have been developed for different intelligent models such as support vector machines [18, 23], neural networks [19], and fuzzy systems [5, 6]. In this study, our focus is transfer learning for fuzzy systems.

Fuzzy systems have been extensively applied in many fields [24, 25]. However, in some fields, it is very difficult to obtain good fuzzy systems without transfer learning. For example, the modeling of the fermentation process [26] is one example where transfer learning is required. In the target domain of a microbiological fermentation process, the data collected may be insufficient or incomplete. A number of missing values are often included. Thus, we cannot effectively model the fermentation process for the target domain with the collected data using traditional fuzzy system modeling methods. However, data available from other similar microbiological fermentation processes could be sufficient. In this situation, the related processes can be considered the source domain for the target domain. Hence, transfer learning can be exploited to make use of the information from the source domain in the modeling for the target domain, which leads to a model with better generalization capability. In this case, transfer learning is an effective solution for the corresponding modeling task because it can enhance the model by leveraging the information available from the source domains, such as the data collected in other time frames or with other setups.

In order to effectively implement the modeling tasks by using fuzzy systems in the abovementioned situation where transfer learning is needed, transfer learning-based fuzzy system modeling methods have been investigated. In [5], a kernel density estimation-based transfer learning mechanism was introduced to develop a modeling method with transfer learning abilities for the Mamdani-Larsen fuzzy system (ML-FS). That is, the knowledge-leverage based ML-FS (KL-ML-FS) modeling method was proposed. In [6], a kind of transfer learning mechanism was proposed for the development of the transfer learning TSK FS modeling method, i.e., the knowledge-leverage based TSK FS (KL-TSK-FS) modeling method. In both the KL-ML-FS and KL-TSK-FS modeling methods, novel objective functions are used to integrate the model knowledge of the source domain and the data of the target domain, and then the fuzzy rules of the model in the target domain are learned with the corresponding optimization techniques. For these knowledge-leveraged based fuzzy system modeling methods, knowledge of the source domain can effectively complement the insufficient data in the target domain. Hence, these methods are very useful in situations where the data are insufficient in the target domain while some useful knowledge of the source domain is available. They can be also viewed as privacy preserving modeling methods since only the knowledge (e.g., the corresponding model parameters) rather than the data of the source domains is utilized.

Between these two transfer learning fuzzy system modelling methods, the KL-TSK-FS has shown a higher degree of flexibility than the KL-ML-FS [5, 6]. This is because the transfer learning mechanism of the KL-TSK-FS is much steadier than the kernel density estimation-based transfer learning mechanism of the KL-ML-FS. Although the KL-TSK-FS has demonstrated promising performance in some applications, there is room for improvement because of the following weaknesses: (1) The antecedent parameters of the TSK-FS model constructed by the KL-TSK-FS are directly inherited from the model obtained in the source domain, making the model that is obtained not particularly appropriate for the modeling task in the target domain. (2) The knowledge-leverage mechanism used for the learning of consequent parameters is still weak. Thus, it is expected that more advanced knowledge-leverage transfer learning mechanisms will be studied.

In order to overcome the abovementioned shortcomings of the KL-TSK-FS modeling method, the enhanced KL-TSK-FS (EKL-TSK-FS) modeling method is investigated in this study. In particular, we proposed the EKL-TSK-FS modeling method from the following two aspects: (1) A transfer fuzzy c-means clustering technique is proposed to realize knowledge-leverage for the antecedents, which enables the antecedent parameter learning to take place simultaneously from the available data in the target domain and from the knowledge of the source domain. (2) An enhanced knowledge-leverage mechanism is introduced for the consequent parameter learning. In addition to the knowledge-leverage term in the original KL-TSK-FS modeling method, another knowledge-leverage term is introduced, which will help the consequent parameters to more efficiently utilize the knowledge from the source domain in the learning procedure. The EKL-TSK-FS has also been extended to a multi-source version for applications where multiple source domains are available.

The remainder of this paper is organized as follows. In Section 2, we briefly explain the concept and principle of classical TSK FS systems and the transfer learning based TSK FS modeling method KL-TSK-FS. In Section 3, the weaknesses of the KL-TSK-FS modeling method are discussed. In Section 4, an enhanced KL-TSK-FS modeling method, i.e. the EKL-TSKFS, is proposed. In Section 5, the EKL-TSKFS is extended for applications in the multi-source scene. The proposed methods are evaluated through computational experiments in Section 6. The conclusions are given in the final section. The initial study for this research on the transfer learning of the TSK FS for the antecedent parameters (i.e., section 5.2) was reported in the Fuzz-IEEE 2014 conference [27].

2. CLASSICAL TSK-TYPE FUZZY SYSTEMS AND THE TRANSFER LEARNING BASED MODELING METHOD

Due to its effectiveness among the classical fuzzy system models, the TSK FS model is the most popular model [28-30]. In this section, the concept and principle of the classical TSK FS and the transfer learning based TSK FS modeling method KL-TSK-FS are described briefly.

2.1 TSK FS

The most commonly used fuzzy inference rules for TSK FSs are defined as follows.

TSK Fuzzy Rule R^k :

$$\text{IF } x_1 \text{ is } A_1^k \wedge x_2 \text{ is } A_2^k \wedge \cdots \wedge x_d \text{ is } A_d^k \ , \qquad (1)$$
$$\text{then } f^k(\mathbf{x}) = p_0^k + p_1^k x_1 + \cdots + p_d^k x_d \ , k = 1, \cdots, K \ .$$

In the above fuzzy rule, A_i^k is a fuzzy subset specified by a membership function on the input variable x_i for the kth rule; K is the number of fuzzy rules, and $\wedge$ is a fuzzy conjunction operator. For TSK FSs, each rule is premised on the input vector $\mathbf{x} = [x_1, x_2, \cdots, x_d]^T$, and maps the fuzzy vector $\mathbf{A}^k = (A_1^k, A_2^k, \cdots, A_d^k)^T$ in the input space $\mathbf{A}^k \subset R^d$ to a varying singleton denoted by $f^k(\mathbf{x})$. The output of the TSK FS can be calculated as follows when we use a multiplicative conjunction, a multiplicative implication, and an additive combination:

$$y^0 = \sum_{k=1}^{K} \frac{\mu^k(\mathbf{x}) f^k(\mathbf{x})}{\sum_{k'=1}^{K} \mu^{k'}(\mathbf{x})} = \sum_{k=1}^{K} \tilde{\mu}^k(\mathbf{x}) f^k(\mathbf{x}) \ , \qquad (2.a)$$

with $\mu^k(\mathbf{x})$ and $\tilde{\mu}^k(\mathbf{x})$ being respectively the fuzzy membership function and the normalized fuzzy membership function associated with the fuzzy vector $\mathbf{A}^k$. The output of the TSK fuzzy model in (2.a) can be rewritten as the following linear regression model [32]:

$$y^o = \mathbf{p}_g^T \mathbf{x}_g \ , \qquad (3.a)$$

where

$$\mathbf{x}_e = (1, \mathbf{x}^T)^T \in R^{d+1}, \qquad (3.b)$$
$$\tilde{\mathbf{x}}^k = \tilde{\mu}^k(\mathbf{x}) \mathbf{x}_e \in R^{d+1}, \qquad (3.c)$$
$$\mathbf{x}_g = ((\tilde{\mathbf{x}}^1)^T, (\tilde{\mathbf{x}}^2)^T, \cdots, (\tilde{\mathbf{x}}^K)^T)^T \in R^{K(d+1)}, \qquad (3.d)$$
$$\mathbf{p}^k = (p_0^k, p_1^k, \cdots, p_d^k)^T \in R^{K(d+1)}, \qquad (3.e)$$
$$\mathbf{p}_g = ((\mathbf{p}^1)^T, (\mathbf{p}^2)^T, \cdots, (\mathbf{p}^K)^T)^T \in R^{K(d+1)} \ . \qquad (3.f)$$

This reformulation shows that the consequent parameter learning in the TSK FS can be viewed as parameter learning in the corresponding linear regression model in the mapping new feature space [6, 26, 32].

2.2 KL-TSK-FS

The KL-TSK-FS modeling method has been proposed in order to implement effective learning of a TSK FS for a situation where transfer learning is required [6]. For the KL-TSK-FS, there are two major sources of information for the learning of a TSK FS model: One is the training data in the target domain, and the other is the knowledge from the source domain. By using these two types of information, the parameter learning of the fuzzy model in the target domain is carried out for the corresponding modeling task.

The use of the following objective function was proposed in the KL-TSK-FS [6]:

$$\min_{\mathbf{p}_g, \xi^+, \xi^-, \varepsilon} f_{L2-TSK-FS} + \lambda (\mathbf{p}_g - \mathbf{p}_{g0})^T (\mathbf{p}_g - \mathbf{p}_{g0})$$
$$f_{L2-TSK-FS} = \frac{1}{N\tau} \sum_{i=1}^{N} ((\xi_i^+)^2 + (\xi_i^-)^2) + \frac{1}{2} (\mathbf{p}_g^T \mathbf{p}_g) + \frac{2}{\tau} \cdot \varepsilon$$

$$s.t.\ \begin{cases} y_i - \mathbf{p}_g^T \mathbf{x}_{gi} < \varepsilon + \xi_i^+ \\ \mathbf{p}_g^T \mathbf{x}_{gi} - y_i < \varepsilon + \xi_i^- \end{cases}, \forall i\ . \tag{4}$$

where $\mathbf{p}_{g0}$ is the knowledge from the source domain. The optimization criterion of the KL-TSK-FS in (4) contains two terms. The first term attempts to learn from the data of the target domain, which is directly inherited from the L2-TSK-FS [26]. The second term intends to learn from the knowledge of the source domains. The parameter λ in Eq. (4) is used to balance the influence of these two terms. Its appropriate value can be determined by using the commonly used cross-validation strategy in machine learning. For more details of the KL-TSK-FS, please refer to [6].

3. WEAKNESSES OF KL-TSK-FS

In this section, we discuss the weaknesses of the KL-TSK-FS modeling method. First, the antecedent parameters of the TSK FS constructed in the target domain have been directly inherited from the model obtained in the source domain in this method. This strategy results in the antecedent parameters being not particularly appropriate for the modeling task in the target domain, since any other information such as the training data in the source domain cannot be used in their learning. Second, only the knowledge from the source domain is utilized in the learning of the consequent parameters through the introduced tem $(\mathbf{p}_g - \mathbf{p}_{g0})^T(\mathbf{p}_g - \mathbf{p}_{g0})$ for the target domain in the KL-TSK-FS. Thus, it seems that the knowledge-leverage learning from the source domain is still insufficient. More knowledge-leveraged terms can be introduced to enhance the learning abilities for the consequent parameters of the target domain.

The above discussions suggest that the KL-TSK-FS modeling method [6] can be further improved. In the next section, an enhanced KL-TSK-FS modeling method, i.e. the EKL-TSK-FS, will be proposed for this purpose.

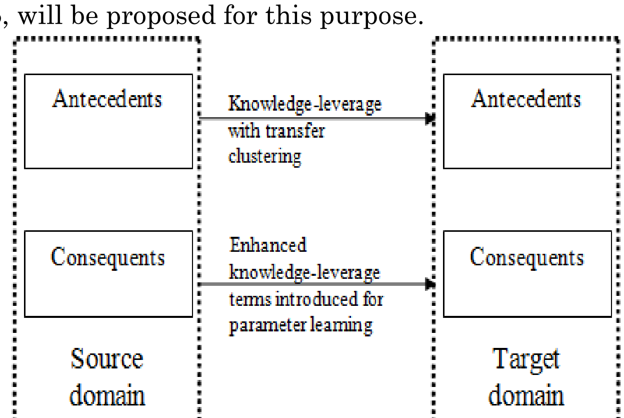

Fig. 2 Knowledge-leverage mechanisms in the proposed EKL-TSK-FS modeling method.

4. EKL-TSK-FS

4.1 Framework of the EKL-TSK-FS

The frameworks of the EKL-TSK-FS are illustrated in Fig. 2. As shown in this framework two issues are to be discussed in the proposed EKL-TSK-FS modeling method: One is the learning of the antecedent parameters based on transfer clustering, and the other is the learning of the consequent parameters based on enhanced knowledge-leverage terms. In the following subsections, the proposed knowledge-leverage mechanisms for the learning of antecedents and consequents are described in detail.

4.2 Enhanced Learning for the Antecedents with Transfer Clustering

In the KL-TSK-FS, the commonly used Gaussian membership functions in the antecedents of the TSK FS include two types of parameters, i.e., the centers c_i^k and the widths δ_i^k , $k=1,\cdots,K$, and $i=1,\cdots,d$. The parameter vectors $\mathbf{c}^k=[c_1^k,\cdots,c_d^k]$, $k=1,\cdots,K$ are obtained as the cluster centers by a fuzzy c-means (FCM) clustering method on the input part of the training dataset [6]. In the KL-TSK-FS, the parameter values of those parameters c_i^k ($k=1,\cdots,K\,;i=1,\cdots,d$) are assumed to be the knowledge available from the source domain and used directly in the TSK FS for the target domain. Thus, these parameter values are not particularly suitable for the target domain since no information in the target domain is used for their learning. In order to overcome this weakness, we propose the following transfer fuzzy c-means (TFCM) clustering technique for the learning of the antecedent parameters of the TSK FS in the target domain.

First, we take the parameter vectors $\mathbf{c}_s^k$, $k=1,\cdots,K$ as the knowledge available from the source domain, which represent the K centers of the fuzzy clusters in the input space of the source domain. Then, we propose a TFCM clustering method to obtain the K centers of the fuzzy partitions in the input space of the target domain using the following objective function:

$$\min_{\mathbf{U},\mathbf{C}_c} J_{TFCM}=\sum\nolimits_{k=1}^{K}\sum\nolimits_{j=1}^{N}u_{kj}^m\left\|\mathbf{x}_j-\mathbf{c}_t^k\right\|^2+\lambda_a\cdot\sum\nolimits_{k=1}^{K}\left(\sum\nolimits_{j=1}^{N}u_{kj}^m\right)\left\|\mathbf{c}_t^k-\mathbf{c}_s^k\right\|^2,$$

$$s.t.\ u_{kj}\in[0,1],\ \ \sum\nolimits_{k=1}^{K}u_{kj}=1,\ \ 1\le j\le N\,. \tag{5}$$

In Eq. (5), $\mathbf{x}_j$ is the *j*th input vector in the training data in the target domain; $\mathbf{c}_c^k$ represents the center of the *k*th fuzzy cluster in the input space of the target domain; u_{kj} denotes the membership value of the *j*th input vector $\mathbf{x}_j$ belonging to the *k*th cluster; $\mathbf{U}=[u_{kj}]_{K\times N}$ and $\mathbf{C}_t=[\mathbf{c}_t^1,\cdots,\mathbf{c}_t^K]$ denote the fuzzy partition matrix and the center matrix, respectively; and λ_a is a balance parameter to control the influence of the two terms in the objective function. The parameter λ_a can be determined by using the commonly used cross-validation strategy.

In particular, we can see that the first term in Eq. (5) is directly inherited from the classical FCM algorithm, which is used to learn the fuzzy partition matrix and the cluster center matrix based on the available data $\mathbf{x}_j$ in the target domain. The second term in Eq. (5) is a knowledge-leverage term, which can be used to learn the cluster centers of the target domain from the knowledge of the source domain.

With Eq. (5) and the optimization technique used in FCM, we can easily obtain the following learning rules for the fuzzy partition matrix and the cluster center matrix:

$$\mathbf{c}_t^k=\frac{\sum\nolimits_{j=1}^{N}u_{kj}^m\mathbf{x}_j+\lambda_a\sum\nolimits_{j=1}^{N}u_{kj}^m\mathbf{c}_s^k}{(1+\lambda_a)\sum\nolimits_{j=1}^{N}u_{kj}^m}=\frac{1}{(1+\lambda_a)}\frac{\sum\nolimits_{j=1}^{N}u_{kj}^m\mathbf{x}_j}{\sum\nolimits_{j=1}^{N}u_{kj}^m}+\frac{\lambda_a}{(1+\lambda_a)}\mathbf{c}_s^k\,, \tag{6}$$

$$u_{kj}=\frac{\left[1\Big/\left(\left\|\mathbf{x}_j-\mathbf{c}_c^k\right\|^2+\lambda_a\left\|\mathbf{c}_t^k-\mathbf{c}_s^k\right\|^2\right)\right]^{1/(m-1)}}{\left[\sum_{k'=1}^{C}\left(1\Big/\left(\left\|\mathbf{x}_j-\mathbf{c}_t^{k'}\right\|^2+\lambda_a\left\|\mathbf{c}_t^{k'}-\mathbf{c}_s^{k'}\right\|^2\right)\right)\right]^{1/(m-1)}} \tag{7}$$

From Eq. (6), we can see that the cluster center $\mathbf{c}_t^k$ that was obtained is written as the sum of the two terms: $(1/1+\lambda_a)\sum_{j=1}^{N}u_{kj}^{m}\mathbf{x}_j/\sum_{j=1}^{N}u_{kj}^{m}$ and $(\lambda_a/1+\lambda_a)\cdot\mathbf{c}_s^k$. The first term can be viewed as the influence of the training data in the target domain, while the second term is the knowledge from the source domain. We can also observe from (7) that the partition matrix is calculated from $\mathbf{x}_j$ in the training data from the target domain and $\mathbf{c}_s^k$ in the knowledge from the source domain.

Once the clustering results are obtained by TFCM in (6) and (7), we can easily calculate the parameter δ_i^k in the antecedent part of the TSK FS accordingly [6].

4.3 Enhanced Learning for the Consequents

In this subsection, we propose an enhanced learning mechanism to improve the knowledge-leverage abilities of the KL-TSK-FS method [6] for the learning of the consequent parameters.

1) Distribution Distance and Maximum Mean Discrepancy (MMD): MMD is a convenient and effective method for measuring the distribution distance. It is a method that has been used in transfer learning for developing some algorithms such as the large-margin projection algorithm (LMPROJ) [35] and the domain adaptation kernelized support vector machine (DAKSVM) [7]. Given a set of M input-output pairs $D_1=\{\{\mathbf{x}_1,y_1\},\ldots,\{\mathbf{x}_n,y_n\}\}$ and a set of m input vectors $D_2=\{\mathbf{z}_1,\ldots,\mathbf{z}_N\}$, the squared MMD distance between the two distributions associated with the two datasets is defined as follows [7, 35]:

$$\mathrm{MMD}^2=\left\|\frac{1}{M}\sum_{j=1}^{n}\phi(\mathbf{x}_j)-\frac{1}{N}\sum_{j=1}^{m}\phi(\mathbf{z}_j)\right\|^2, \tag{6}$$

where $\phi(\boldsymbol{x}_j)$ is a mapping function.

In this study, it is used to enhance the inductive KL-TSK-FS modeling method. The goal of a transfer learning method with an MMD mechanism is usually to find a projected vector that minimizes the distance between two distributions in the projected space, while at the same time optimizing the performance of the model for the training data. Given a dataset $D_s=\{\{\mathbf{x}_{1,r},y_{1,r}\},\ldots,\{\mathbf{x}_{M,r},y_{M,r}\}\}$ in the source domain and a training data set in the target domain $D_t=\{\{\mathbf{x}_1,y_1\},\ldots,\{\mathbf{x}_N,y_N\}\}$ for our TSK FS modeling task, we can obtain the following two datasets by using the mapping in Eqs. (3.b)-(3.d) with the fixed antecedents of the TKS-FS learned by the TFCM clustering in Eq. (5):

$$D_{s,map}=\left\{(\mathbf{x}_{g1,r},y_{1,r}),\ldots,(\mathbf{x}_{gM,r},y_{M,r})\right\} \tag{7.a}$$

$$D_{t,map}=\{\{\mathbf{x}_{g1},y_1\},\ldots,\{\mathbf{x}_{gN},y_N\}\}. \tag{7.b}$$

Then, MMD in (6) can be defined as follows to estimate the distribution distance between the source domain and the target domain under an expected projection $\mathbf{p}_g$ for the TSK FS model in the target domain:

$$\begin{aligned}d(P_{s,map},P_{t,map})&=\mathrm{MMD}^2=\left\|\frac{1}{N}\sum_{i=1}^{N}\mathbf{p}_g^T\mathbf{x}_{gi}-\frac{1}{M}\sum_{i=1}^{M}\mathbf{p}_g^T\mathbf{x}_{gi,r}\right\|^2\\&=\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\mathbf{p}_g^T\mathbf{x}_{gi}\mathbf{x}_{gj}^T\mathbf{p}_g+\frac{1}{M^2}\sum_{i=1}^{M}\sum_{j=1}^{M}\mathbf{p}_g^T\mathbf{x}_{gi,r}\mathbf{x}_{gj,r}^T\mathbf{p}_g-\frac{2}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}\mathbf{p}_g^T\mathbf{x}_{gi}\mathbf{x}_{gj,r}^T\mathbf{p}_g\end{aligned} \tag{8}$$

where $P_{s,map}$ and $P_{t,map}$ denote the two distributions of mapping data in the new feature space, for the source domain and the target domain, respectively. Let

$$\boldsymbol{\Omega}_0 = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\mathbf{x}_{gi}\mathbf{x}_{gj}^T + \frac{1}{M^2}\sum_{i=1}^{M}\sum_{j=1}^{M}\mathbf{x}_{gi,r}\mathbf{x}_{gj,r}^T - \frac{2}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}\mathbf{x}_{gi}\mathbf{x}_{gj,r}^T \ , \tag{9.a}$$

$$\boldsymbol{\Omega} = \left(\boldsymbol{\Omega}_0 + \boldsymbol{\Omega}_0^T\right)/2 \ . \tag{9.b}$$

Then (8) can be expressed as

$$d(P_{s,map}, P_{t,map}) = \mathrm{MMD}^2 = \mathbf{p}_g^T\boldsymbol{\Omega}\mathbf{p}_g \ . \tag{10}$$

2) Enhanced Objective Function: We propose the following objective function by introducing the projected squared MMD distance in Eq. (15) to enhance the knowledge-leverage abilities of the KL-TKS-FS [6]:

$$\begin{aligned} &\min_{\mathbf{p}_g,\xi^+,\xi^-,\varepsilon} \ f_{L2-TSK-FS} + \lambda_1(\mathbf{p}_g - \mathbf{p}_{g0})^T(\mathbf{p}_g - \mathbf{p}_{g0}) + \lambda_2 d(P_{s,map}, P_{t,map}) \\ &f_{L2-TSK-FS} = \frac{1}{N\tau}\sum_{i=1}^{N}((\xi_i^+)^2 + (\xi_i^-)^2) + \frac{1}{2}(\mathbf{p}_g^T\mathbf{p}_g) + \frac{2}{\tau}\cdot\varepsilon \\ &s.t. \ \begin{cases} y_i - \mathbf{p}_g^T\mathbf{x}_{gi} < \varepsilon + \xi_i^+ \\ \mathbf{p}_g^T\mathbf{x}_{gi} - y_i < \varepsilon + \xi_i^- \end{cases} . \end{aligned} \tag{11}$$

where $d(P_{s,map}, P_{t,map})$ is defined in Eq. (8) or (10). From Eq. (11), the following observations can be made.

(1) The first and second terms are inherited directly from the KL-TSL-FS and used to learn from the data in the target domain and the model parameters in the source domain simultaneously.

(2) The third term is introduced to measure the distribution distance between the source domain and the target domain, which is expected to enhance the ability of the KL-TSK-FS to learn from the source domain.

3) Optimization of the Enhanced Objective Function: It is no trivial matter to solve Eq. (11) directly. This optimization problem is often transformed into a more easily solved dual problem. The dual problem of Eq. (11) can be formulated as the following QP problem:

$$\begin{aligned} &\min_{\boldsymbol{\alpha}^+,\boldsymbol{\alpha}^-} \ -\frac{1}{2}\left\{\frac{N\tau}{2}\sum_{i=1}^{N}\left((\alpha_i^+)^2 + (\alpha_i^-)^2\right) + \sum_{i=1}^{N}\sum_{j=1}^{N}\left(\alpha_i^+ - \alpha_i^-\right)\left(\alpha_j^+ - \alpha_j^-\right)\cdot h_{ij}\right\} + \sum_{i=1}^{N}(\alpha_i^+ - \alpha_i^-)\left(y_i + b_i\right), \\ &s.t. \quad \alpha^+ \geq 0, \ \alpha^- \geq 0 \ , \ \sum_{i=1}^{N}\left(\alpha_i^+ + \alpha_i^-\right) = \frac{2}{\tau} \ . \end{aligned} \tag{12.a}$$

with

$$h_{ij} = \mathbf{x}_{gi}^T\left(2\left(\mathbf{R}^{-1}\right)^T - (1+2\lambda_1)(\mathbf{R}^{-1})^T(\mathbf{R}^{-1}) - 2\lambda_2\left(\mathbf{R}^{-1}\right)^T\boldsymbol{\Omega}\mathbf{R}^{-1}\right)\mathbf{x}_{gj} \ , \tag{12.b}$$

$$b_i = \left\{2\lambda_1\left(\mathbf{p}_{g0}\right)^T(\mathbf{R}^{-1})^T(\mathbf{R}^{-1}) + 4\lambda_1^2\left(\mathbf{p}_{g0}\right)^T\left(\mathbf{R}^{-1}\right)^T\mathbf{R}^{-1} - 2\lambda_1\left(\mathbf{p}_{g0}\right)^T\mathbf{R}^{-1} + 2\lambda_1\lambda_2\left(\mathbf{p}_{g0}\right)^T\left(\mathbf{R}^{-1}\right)^T\left(\boldsymbol{\Omega}^T + \boldsymbol{\Omega}\right)\mathbf{R}^{-1} - 2\lambda_1\left(\mathbf{p}_{g0}\right)^T\left(\mathbf{R}^{-1}\right)^T\right\}\mathbf{x}_{gi} \tag{12.c}$$

$$\mathbf{R} = ((1+2\lambda_1)\mathbf{I}_{K(d+1)\times K(d+1)} + 2\lambda_2\boldsymbol{\Omega}^T) \in R^{K(d+1)\times K(d+1)} \ . \tag{12.d}$$

In (12.a), $\boldsymbol{\alpha}^+, \boldsymbol{\alpha}^-$ are the Lagrangian multiplier vectors, i.e., the solution variables of the dual problem of (11). The derivation of (12.a) can be seen in Appendix 1.

According to the KTT optimal theory, the optimal consequent parameters of the trained TSK FS in the target domain, i.e. $(\mathbf{p}_g)^*$ in (11), can finally be given by

$$\mathbf{p}_g = \mathbf{R}^{-1}\left(2\lambda_1\mathbf{p}_{g0} + \sum_{i=1}^{N}(\alpha_i^+ - \alpha_i^-)\mathbf{x}_{gi}\right) \tag{13}$$

where $(\alpha_i^+)^*, (\alpha_i^-)^*$ are the optimal solutions of the dual problem in (12.a). The derivation of (13) can also be seen in Appendix 1.

For (12.a), we can give a more compact form, as follows:

$$\max_{\boldsymbol{\upsilon}} \quad -\frac{1}{2}\boldsymbol{\upsilon}^{\mathrm{T}}\mathbf{H}\boldsymbol{\upsilon} + \boldsymbol{\upsilon}^{\mathrm{T}}\boldsymbol{\gamma}$$

$$s.t. \;\; \boldsymbol{\upsilon}^{\mathrm{T}}\mathbf{1} = \frac{2}{\tau}, \quad \boldsymbol{\upsilon}_i \geq 0 \;\; \forall i, \tag{14}$$

where

$$\boldsymbol{\upsilon} = \left[(\boldsymbol{\alpha}^+)^T, (\boldsymbol{\alpha}^-)^T\right]^T, \tag{15.a}$$

$$\boldsymbol{\gamma} = \left[\left(\mathbf{y}+\mathbf{b}\right)^T, \; -\left(\mathbf{y}+\mathbf{b}\right)^T\right]^T, \tag{15.b}$$

$$\mathbf{b} = \left(b_1, \cdots, b_n\right)^T, \tag{15.c}$$

$$\mathbf{H} = \begin{pmatrix} \mathbf{K} & -\mathbf{K} \\ -\mathbf{K} & \mathbf{K} \end{pmatrix}, \tag{15.d}$$

$$\mathbf{K} = [\tilde{k}_{ij}]_{N\times N}, \;\; \tilde{k}_{ij} = h_{ij} + \frac{N\tau}{2}\delta_{ij}, \tag{15.e}$$

$$\delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}. \tag{15.f}$$

4.4 Algorithm of EKL-TSK-FS

Based on the abovementioned two learning mechanisms of the EKL-TSK-FS in Subsections 4.2 and 4.3, the overall learning algorithm of the proposed EKL-TSK-FS modeling method can be written as follows.

Algorithm for the EKL-TSK-FS	
Step 1	Introduce the knowledge of the source domain, i.e., the model parameters.
Step 2	Set the balance parameters λ_a in (5) and $\tau, \lambda_1, \lambda_2$ in (11).
Step 3	Use (5)-(7) to learn the antecedent parameters of the TSK FS in the target domain.
Step 4	Use (3.b)-(3.d) and the antecedent parameters of the TSK FS of the target domain to construct the datasets $D_{s,map}$ and $D_{t,map}$ in (7.a) and (7.b), respectively.
Step 5	Use (12.a) or (14) to obtain the consequent parameters of the TSK FS in the target domain.

Remarks: Compared with the KL-TSK-FS algorithm, the advantage of the EKL-TSK-FS is that the enhanced transfer learning mechanism is used. However, a disadvantage is also introduced due to the learning mechanism that was adopted. We know that only the model parameters in the source domain are used in the KL-TSK-FS algorithm, giving it better privacy protection performance. Since the mapping dataset, i.e. $D_{s,map}$ in (7.a), is required in the EKL-TSK-FS algorithm, the privacy protection performance of the EKL-TSK-FS is weaker than that of the KL-TSK-FS.

The computational complexity of the EKL-TSK-FS algorithm is analyzed briefly here. It has the following two main parts: 1) the learning of the antecedent parameters; and 2) the learning of the consequent parameters. For the first part, since the antecedent parameters are obtained by the TFCM clustering technique, its computational complexity is equal to that of the classical FCM clustering algorithm, i.e. O(KNT), where K, N and T denote the number of clusters, the number of training data, and the number of iterations, respectively. For the second part, the consequent parameters are obtained by solving the QP problem in Eq. (12.a); thus, its complexity is usually O(N^2) of typical QP problems. However, if some sophisticated QP algorithms are adopted, such as the working set-based algorithm [36], the computational complexity can be reduced to between O(N) and O(N^2), depending on the QP solutions used. In this study, we have adopted the working set-based QP solution [36] for solving the QP problem in the experimental studies.

5. EXTENDED EKL-TSK-FS FOR MULTI-SOURCE DOMAINS

Although the proposed EKL-TSK-FS can improve the performance of the KL-TSK-FS [6], like the KL-TSK-FS it is only applicable for the scene of a single source domain. In the real world, it is common for several sources domains to be available for a target domain. In this section, the EKL-TSK-FS is extended for the above situation, and a multi-source domain based EKL-TSK-FS method (MS-EKL-TSK-FS) is proposed.

5.1 Framework of the MS-EKL-TSK-FS

For a multi-source domain scene, the expectation is that through collaborative learning and adaptive learning multiple source domains can improve the performance of the TSK FS that was obtained for the target domain. Through collaborative learning, multi-source domains can benefit from each other, which will be useful for the learning in the target domain. With adaptive learning, the influence of different source domains can be adjusted adaptively. Based on the above idea, a learning framework for the MS-EKL-TSK-FS is presented in Fig. 3. Based on this framework, an ensemble TSK FS will be obtained for the modeling task in the target domain.

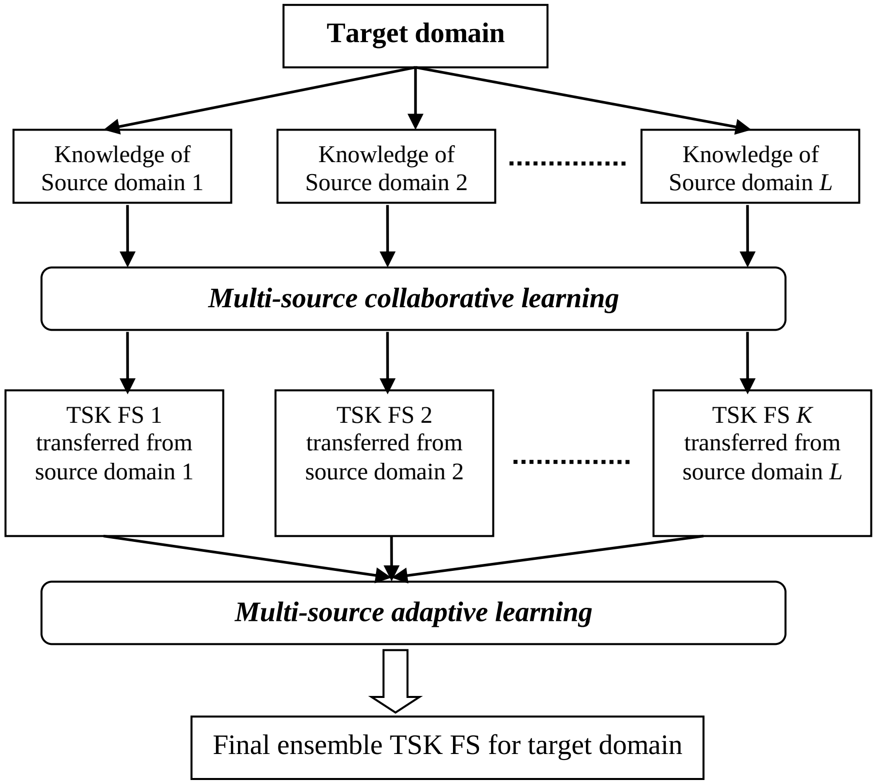

Fig. 3 The framework of the MS-EKL-TSK-FS.

5.2 Two learning mechanisms and the objective function for multi-source domain scene

Based on the above framework, the following two learning mechanisms are introduced for the proposed MS-EKL-TSK-FS: (1) Multi-source adaptive learning, and (2) Multi-source collaborative learning. The former is used to adaptively adjust the influence of different source domains on the target domain, while the latter is used to realize the collaborative learning of different source domains and further boost the learning effect for the target domain. Details of the two mechanisms are given below.

1) Multi-source adaptive learning: In order to realize multi-source adaptive learning, the following objective is proposed:

$$\begin{aligned}\min_{\mathbf{p}_{g,l},w_l} J_{\text{MS-ADP}} = & \frac{1}{2}\sum_{l=1}^{L} w_l \sum_{i=1}^{N} \left\| \left(\mathbf{p}_{g,l}\right)^T \mathbf{x}_{gi,l} - y_i \right\|^2 + \lambda_1 \sum_{l=1}^{L} w_l \ln w_l \\ & + \frac{\lambda_2}{2}\sum_{l=1}^{L}\left(\mathbf{p}_{g,l}\right)^T \mathbf{p}_{g,l} + \lambda_3 \sum_{l=1}^{L}(\mathbf{p}_{g,l}-\mathbf{p}_{g0,l})^T(\mathbf{p}_{g,l}-\mathbf{p}_{g0,l}) + \lambda_4 \sum_{l=1}^{L}\mathbf{p}_{g,l}^T \mathbf{\Omega}_l \mathbf{p}_{g,l} \end{aligned} \tag{16}$$

$$s.t. \quad \sum_{l=1}^{L} w_l = 1 .$$

In (16), the first term is the weighted least-squares (LS) penalty terms for measuring the errors of the trained L TSK FSs by transferring knowledge from different source domains. To easily solve the objective, the LS penalty terms in (16) have been used instead of the ε-insensitive loss penalty terms used in the EKL-

TSK-FS for single source learning. The second term is the negative Shannon entropy, which has been extensively used for the adaptive learning in many algorithms [52, 53]. By integrating the first and second terms, the importance of different source domains to the target domain can be adjusted by learning the weights involved in the first two terms of (16). In (16), the last three terms are directly inherited from the single source learning method, i.e., the EKL-TSK-FS.

2) Multi-source collaborative learning: Since several source domains are available, we hope that these source domains may realize collaborative learning to further improve the learning effect of the target domain. For this purpose, the following objective function is proposed:

$$\min_{\mathbf{p}_{g,l},w_l} J_{\text{MS-COL}} = \frac{1}{2}\sum_{l=1}^{L}\sum_{i=1}^{N}\left\|\left(\mathbf{p}_{g,l}\right)^T \mathbf{x}_{\text{g}i,l} - \frac{1}{L-1}\sum_{h=1,h\neq l}^{L}\left(\tilde{\mathbf{p}}_{g,h}\right)^T \mathbf{x}_{\text{g}i,h}\right\|^2 \tag{17}$$

In (17), when $\mathbf{p}_{g,l}$, which stands for the consequent parameters that have been transferred from the lth source domain, is to be optimized for the target domain, $\tilde{\mathbf{p}}_{g,h}$ $(h\neq l)$, which denotes the consequent parameters that have been transferred from the other source domains, will be used to instruct the learning from the lth source domain. By using (17), the intention is for different source domains to help the target domain obtain consistent prediction results by learning from different source domains as far as possible.

3) Objective function for multi-source domain scene: Based on the above two multi-source transfer learning mechanisms, a final objective function for the MS-EKL-TSK-FS is designed as follows:

$$\min_{\mathbf{p}_{g,l},w_l} J_{\text{MS-EKL-TSK-FS}} = J_{\text{MS-ADP}} + \lambda_5 J_{\text{MS-COL}}\ . \tag{18}$$

By optimizing (18), the parameters $\mathbf{p}_{g,l}$ $(l=1,\cdots,L)$, which correspond to the L TSK FS that are transferred from L source domains, will be obtained along with the their weights w_k. The final decision can then be given in an ensemble learning way, as follows:

$$f(\mathbf{x}_i) = \sum_{l=1}^{L} w_l \left(\mathbf{p}_{\text{g},l}\right)^T \mathbf{x}_{\text{g}i,l} \tag{19}$$

Since the objective function in (18) is a non-convex problem, solving it is no trivial matter. Here, an alternate learning strategy is used to optimize (18). The main learning rules are given as follows:

$$\begin{aligned}\mathbf{p}_{g,l} = &\left((w_l+\lambda_5)\sum_{i=1}^{N}\left(\mathbf{x}_{\text{g}i,l}\right)^T \mathbf{x}_{\text{g}i,l} + (\lambda_2+2\lambda_3)\mathbf{I}_{d_{g,l}\times d_{g,l}} + 2\lambda_4 \mathbf{\Omega}_l^{\ T}\right)^{-1} \\ &\cdot\left(w_l\sum_{i=1}^{N}\mathbf{x}_{\text{g}i,l}y_i + \frac{\lambda_5}{L-1}\sum_{h=1,h\neq l}^{L}\sum_{i=1}^{N}\mathbf{x}_{\text{g}i,h}\left(\tilde{\mathbf{p}}_{g,h}\right)^T \mathbf{x}_{\text{g}i,h} + 2\lambda_3 \mathbf{p}_{g0,l}\right)\end{aligned} . \tag{20}$$

$$w_l = \frac{\exp\left(-\frac{1}{2}\sum_{i=1}^{N}\left\|\left(\mathbf{p}_{g,l}\right)^T \mathbf{x}_{\text{g}i,l} - y_i\right\|^2 \Big/ \lambda_1\right)}{\sum_{h=1}^{L}\exp\left(-\frac{1}{2}\sum_{i=1}^{N}\left\|\left(\mathbf{p}_{g,h}\right)^T \mathbf{x}_{\text{g}i,h} - y_i\right\|^2 \Big/ \lambda_1\right)} . \tag{21}$$

In (20), $\mathbf{I}_{d_{g,l}\times d_{g,l}}$ is an identity matrix of $d_{g,l}\times d_{g,l}$ where $d_{g,l}$ is the dimensionality of $\mathbf{x}_{gi,l}$, and $\mathbf{\Omega}_l$ is defined as in (9.b) for solving $\mathbf{p}_{g,l}$. For detailed derivations of (20) and (21), please see Appendix 2. Based on the above analyses, the algorithm of the MS-EKL-TSK-FS can be obtained. Details of the algorithms are given in Appendix 3.

6. EXPERIMENTAL STUDIES

6.1 Experimental Settings

The proposed EKL-TSK-FS is evaluated on both synthetic and real-world datasets by comparing it with the related methods. For clarity, the notations of the adopted datasets and their definitions are listed in Table 1, and the adopted algorithms for comparison are delineated in Table 2. The generalization performance index *J* in (22) is used in our experiments [33],

$$J=\sqrt{\frac{\sum_{i=1}^{N}(y_i'-y_i)^2\Big/N}{\sum_{i=1}^{N}(y_i-\bar{y})^2\Big/N}}, \tag{22}$$

where *N* is the number of test data; y_i is the output for the *i*th test input; y_i' is the fuzzy model output for the *i*th test input; and $\bar{y}=\sum_{i=1}^{N}y_i\Big/N$. The smaller the value of *J*, the better the generalization performance.

In our experiments, a five-fold cross-validation strategy was used to determine the hyper parameters for all of the algorithms that were adopted based on the training datasets. All of the algorithms are implemented using MATLAB on a computer with an Intel Core 2 Duo P8600 2.4 GHz CPU and 2GB of RAM.

Table 1 Notations of the adopted datasets with their definitions

Notation	Definitions
D1	Dataset generated in the source domain.
D2	Dataset generated in the target domain for training.
D2_test	Dataset generated in the target domain for testing.
r	Relation parameter between the source domain and the target domain used to construct the synthetic datasets.

Table 2 The methods used for performance comparisons

Algorithm	Descriptions
L2-TSK-FS (D1) [26]	L2-TSK-FS trained by the data in the source domain.
L2-TSK-FS (D2) [26]	L2-TSK-FS trained by the data in the target domain.
L2-TSK-FS (D1+D2) [26]	L2-TSK-FS trained by the data in both the target domain and the source domain.
ANFIS(D2) [37]	An adaptive neuro-fuzzy inference systems training method of the Sugeno-type fuzzy system in the target domain.
GENFIS2(D2) [37]	A Sugeno-type fuzzy inference systems training method using subtractive clustering in the target domain.
TSFS-SVR(D2) [38]	TS-fuzzy system-based support vector regression in the target domain.
FS-FCSVM (D2) [39]	Fuzzy system learned through fuzzy clustering and a support vector machine in the target domain.
TrAdaBoost (D1+D2) [40]	Transfer AdaBoost based on the LS-SVR learner with the RBF-type kernel function for regression by the data in both the target domain and the source domain.
HiRBF (D1+D2) [19]	Bayesian task-level transfer learning for the non-linear regression method using the data in both the target domain and the source domain.
KL-TSK-FS (D2+Knowledge) [6]	Knowledge-leverage based TSK fuzzy system.
EKL-TSK-FS (D2+Knowledge)	The proposed enhanced KL-TSK-FS.
MS-EKL-TSK-FS (D2+Knowledge)	The proposed enhanced KL-TSK-FS for multiple source domains.

6.2 Synthetic Datasets

1) Generation of Synthetic Datasets: Synthetic datasets are constructed to cater to the scene studied in this paper with the following procedure: First, the function $Y = x * \sin(x),\ x \in [-10,10]$ is adopted to represent the source domain and to generate dataset D1. Second, the function $y = x * \sin(x) + r * x,\quad x \in [-10,10]$ is used to describe the target domain and to generate dataset D2 and D2_test as the training and testing datasets in the target domain. We use r as a relation parameter between the source domain and the target domain to control the degree of similarity/difference between these two domains. When r =0, the two domains are identical. In particular, the lack of information (data insufficiency) is simulated by introducing intervals with missing data into the training set for the target domain. The settings to generate the synthetic datasets have been given in Table 3.

The two related domains that were simulated, using the relation parameter $r = 0.7$, are shown in Fig. 4(a). Meanwhile, based on the scene in Fig. 4(a), the data in the source domain and the training data in the target domain are shown in Fig. 4(b), along with the two intervals that are missing data, [-7, -4] and [3, 6], introduced in the target domain.

(a) (b)

Fig. 4 Functions representing two different domains with the relation parameter $r = 0.7$ and the corresponding sampling data from these domains: (a) functions representing the source domain (Y) and the target domain (y); (b) data in the source domain and the training data in the target domain with missing data in the intervals [-7,-4] and [3,6].

Table 3 Details of the synthetic datasets

Source domain	Target domain		
Dataset	Training set		Testing set
Size of dataset	Interval with missing data	Size of dataset	Size of dataset
400	[-7,-4] and[0,3]	140	200
Relation parameter between the two domains: r =0.3, 0.5, 0.7and 0.9.			

2) *Performance Comparison*: In order to make a statistically meaningful comparison, experiments were conducted by taking samples from the source domain and the target domain, 80% of which were random. Each experiment was repeated 10 times, and the statistical results with means and standard deviations were reported. To conduct an in-depth evaluation of the effectiveness of the proposed EKL-TSK-FS and the corresponding enhanced knowledge-leverage mechanisms, comparisons from four aspects were designed as follows:

(i) The performance of the existing L2-TKS-FS-based methods and the proposed EKL-TSK-FS was compared.

(ii) The effectiveness of the enhanced knowledge-leverage mechanisms was compared, i.e., enhanced learning for antecedents with transfer clustering and enhanced learning for consequents with MMD technology.

(iii) The performance of the related non-transfer learning and transfer learning methods were compared.

(iv) To evaluate the influence of the size of the training data in the target domain, the corresponding results that were obtained based on different percentages of data in D2 were compared.

The results of different methods on the synthetic datasets are compared in Tables A1-A4 in Appendix 4. Detailed analyses of the above four aspects are presented below.

(1) In Table A1, the generalization performance of the proposed EKL-TSK-FS method is obviously better than that of the two non-transfer L2-TSK-FS methods, i.e., L2-TSK-FS(D1) and L2-TSK-TSK(D2). In addition, compared with the L2-TSK-FS(D1+D2), although the L2-TSK-FS was trained using both the data D2 in the target domain and the data D1 in the source domain, it was still unable to achieve better performance than the proposed EKL-TSK-FS. The above results tell us that the traditional L2-TSK-FS modeling method using incomplete datasets will suffer from weak generalization due to the absence of transfer learning abilities. Although the existing KL-TSK-FS has the ability to carry out transfer learning, its performance is weaker than the proposed EKL-TSK-FS method. The results reveal that the proposed EKL-TSK-FS with two enhanced knowledge-leverage mechanisms can further enhance the ability to carry out transfer learning when training a TSK FS model.

(2) In order to evaluate the effectiveness of the two enhanced knowledge-leverage mechanisms of EKL-TSK-FS that were introduced, different combinations of strategies have been adopted to develop the TSK FS training methods. As shown in Table A2 in Appendix 4, FCM+KL-TSK-FS is the TSK FS knowledge-leverage transfer learning method in [6]; TFCM+KL-TSK-FS denotes the TSK FS transfer learning method using TFCM for the antecedents and the knowledge-leverage transfer learning mechanism in [6] for the consequents; FCM+MMD-KL-TSK-FS denotes the TSK FS transfer learning method using FCM for the antecedents in [6] and the proposed MMD based knowledge-leverage transfer learning mechanism for the consequents; EKL-TSK-FS used the proposed TFCM and the MMD based knowledge-leverage transfer learning mechanisms for the antecedents and the

consequents, respectively. Through the results in Table A2 in Appendix 4, we can clearly evaluate the effectiveness of the two knowledge-leverage mechanisms that were introduced. First, comparing the results between the third column and the fourth column in Table A2, we can see that the transfer clustering mechanism can cause the antecedents to have transfer learning abilities, accordingly improving the performance of the KL-TSK-FS. Similar results can be found by comparing the results of the third and fifth columns in Table A2. The corresponding results reveal that the MMD technology that was introduced enhanced the transfer learning abilities of the consequents in [6], compared with the previous knowledge-leverage mechanism. Finally, we can see that the best generalization performance appeared in the sixth column of Table A2, indicating that a TSK FS trained by simultaneously using the two knowledge-leverage mechanisms that were introduced can further improve the ability of transfer learning.

(3) Comparing the results between the several non-transfer learning and transfer learning methods in Table A3 in Appendix 4, we find that in most cases the generalization performance of the transfer learning methods was better than that of the non-transfer learning methods. In particular, when the two knowledge-leverage based transfer learning methods, i.e. the KL-TSK-FS and the EKL-TSK-FS, are compared with other transfer learning methods, they demonstrated better generalization capability in the data-missing scene due to their knowledge-leveraged abilities. In addition, by further comparing the KL-TSK-FS with the EKL-TSK-FS, we can see from the performance index reported in Table A3 that the EKL-TSK-FS demonstrated stronger knowledge-leverage abilities than the KL-TSK-FS. The above observations also confirm that the proposed enhanced knowledge-leverage mechanism in the EKL-TSK-FS is better than that used in the KL-TSK-FS.

(4) The corresponding results obtained with different percentages of data in D2 (r =0.5) used for model training are listed in Table A4 in Appendix 4. In this experiment, the other settings are the same as in the other experiments analyzed above. The results show that as the size of the training data used in D2 increased, the performance of the proposed method improved accordingly. Compared with other existing related methods, the proposed method also demonstrates better or competitive performance. When the size of the training data used in D2 is small, the negative influence of the source domain is still avoided since the cross-validation strategy that was adopted can determine the appropriate balance parameters to effectively weaken the negative influence of the source domain.

6.3 Real-world Datasets: Fermentation Process modeling Datasets

1) Glutamic Acid Fermentation Process Modeling: The proposed method is further evaluated with a real-world dataset [5, 6, 26], by using it to model a biochemical process. The dataset that was adopted was collected from the glutamic acid fermentation process. This dataset includes six input variables, i.e., the fermentation time h, the glucose concentration $S(h)$, the thalli concentration $X(h)$, the glutamic acid concentration $P(h)$, the stirring speed $R(h)$, and the ventilation $Q(h)$, with $h=0,\ 2,\ \cdots,28$. The output variables of this data contain the glucose concentration $S(h+2)$, the thalli concentration $X(h+2)$, and the glutamic acid concentration $P(h+2)$ at a future time $h+2$. The TSK FS model based on the biochemical process prediction model is illustrated in Fig. 5. All of the data in this dataset were collected from 21 batches of fermentation processes, where each batch contained 14 effective input-output data samples. In our experiment, in order to match the situation discussed in this study, the data are divided into two domains, i.e. the source domain and the target domain, as described in Table 4.

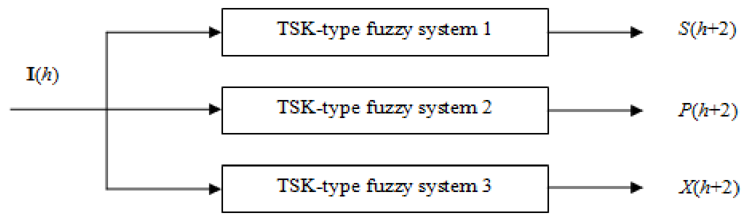

Fig. 5 Illustration of the glutamic acid fermentation process prediction model based on the TSK FSs.

Table 4 The fermentation process modeling dataset

	Data of the source domain (D1)	Data of the target domain	
		Training set (D2)*	Testing set (D2_test)
Number of batches	16	3	2
Size of the dataset	224	30	28

*For training set of the target domain, information is missing at time h = 6, 8, 10, 12.

Table 5 Generalization performance (J) of the non-transfer and transfer learning methods in fermentation process modeling

Part A: Non-transfer Learning Methods							
Output	Index	ANFIS (D2)	GENFIS2 (D2)	TSFS-SVR (D2)	FS-FCSVM (D2)	L2-TSK-FS (D1)	L2-TSK-FS (D2)
$S(h+2)$	mean	0.7129	0.5593	0.5536	0.5308	0.3068	0.5036
	std	0.1525	0.0706	0.0445	0.0578	0.0350	0.0669
$X(h+2)$	mean	3.9366	1.2386	1.3084	1.1637	0.7323	1.3718
	std	1.2685	0.1326	0.2632	0.1245	0.0746	0.2607
$P(h+2)$	mean	0.6451	0.4096	0.3806	0.3493	0.2563	0.3667
	std	0.2482	0.1063	0.0793	0.0537	0.0567	0.0511
Part B: Transfer Learning Methods							
Output	Index	L2-TSK-FS (D1+D2)	TrAdBoos t (D1+D2)	HiRBF (D1+D2)	KL-TSK-FS (D2+Knowledge)	EKL-TSK-FS (D2+Knowledge)	
$S(h+2)$	mean	0.3898	0.3436	0.3171	0.1488	**0.0824**	
	std	0.0368	0.0534	0.0424	0.0495	**0.0142**	
$X(h+2)$	mean	0.9077	0.7703	0.7593	0.4891	**0.3413**	
	std	0.0977	0.0247	0.0302	0.0129	**0.0321**	
$P(h+2)$	mean	0.3088	0.2989	0.3276	0.1626	**0.1206**	
	std	0.0476	0.0246	0.0472	0.0243	**0.0136**	

2) Performance Comparison: In order to make a statistically meaningful comparison, experiments based on the random partitioning of 21 batches of data with a ratio of 16:3:2 for the source domain, the target domain (training), and the target domain (test) were implemented. The statistical results are reported in Table 5.

Table 5 shows that the results of the modeling of the EKL-TSK-FS method are better than those of the other methods. Again, this can be explained by the fact that the proposed method can effectively exploit not only the data in the target domain but also useful knowledge from the source domain in the training procedure for the modeling task in the target domain. Even if the data in the target domain are insufficient for training, the generalization capability of the TSK FS model obtained by the proposed EKL-TSK-FS method is not significantly degraded. Since a lack of data is becoming increasingly common due to the poor sensitivity of sensors in noisy environments, the above remarkable capacity of the EKL-TSK-FS is very valuable for biochemical process modeling. From the results in Table 5, we can see that although the KL-TSK-FS method also has knowledge-leverage abilities, its generalization abilities are weaker than those of the proposed EKL-TSK-FS method due to its insufficient knowledge-leverage learning. Thus, the EKL-TSK-FS is more promising than the KL-TSK-FS for the practical application of fermentation process modeling.

In Fig. A1 of Appendix 5, we also show the modeling effect of different methods in a certain run for a batch of test data. In particular, for the same batch of test data, the modeling effect of the adopted algorithms is shown in two subfigures, so as to make the modeling effect more visible. As shown in Fig. A1, the modeling effect for the output of S(h+2) on a certain batch of data in the test dataset was demonstrated using both Fig. A1(a) and Fig. A1(b). From Fig. A1 in Appendix 5 it seems that the performance of the proposed method is close to that of some existing methods. This may be due to the fact that the practical application is more complicated and, thus, the visual evaluation may not be objective. Indeed, from the objective quantitative indices in Table 5 we find that the proposed method achieved better performance.

6.4 High Dimensional Mortgage Dataset

A real-world mortgage dataset (http://funapp.cs.bilkent.edu.tr/DataSets/) was adopted to evaluate the performance of the proposed EKL-TSK-FS in the high dimensional dataset. Details of this experiment, and its results, can be found in Appendix 6. From the results reported in Table A5 in Appendix 6, we find that for high dimensional datasets, the proposed method still achieves a promising performance. For this observation, we want to give the following explanations: although many classical TSK FS modeling algorithms, such as ANFIS, are not suitable for high dimensional datasets, the challenge has to some extent been overcome in many recently developed TSK FS modeling methods. For example, in many methods the structural risk minimization technique has been introduced to improve the generalization ability of the trained TSK FS. In this study, the proposed EKL-TSK-FS also inherited this ability, and thus demonstrated promising performance in the high mortgage dataset.

6.5 Comparison of Computation Cost

In this subsection, the computational cost of the adopted methods is compared in several datasets. The experimental results are listed in Table A6 of Appendix 7. The experimental results show that the computation cost of the proposed EKL-TSK-FS falls in the middle of the ten adopted methods.

6.6 Performance Evaluation in Multi-source Domain Scenes

In this subsection, the performance of the MS-EKL-TSK-FS, which is an extended version of the EKL-TSK-FS for multi-source scenes, is evaluated on several multi-source domain datasets. Here, only the results in the fermentation process modeling multi-source domain dataset are reported. More experimental results on other multi-source datasets can be seen in Appendix 8. Details of the multi-source fermentation process modeling dataset are given in Table 6.

Table 6 Constructed multi-source domain fermentation process modeling dataset

	Data of two source domains		Data of the target domain	
	D1-1	D2-2	Training set (D2)*	Testing set (D2_test)
Number of batches	8	8	3	2
Size of the dataset	112	112	30	28

*For the training set of the target domain, information is missing at time h = 6, 8, 10, 12.

To make a statistically meaningful comparison, the 21 batches of data were randomly partitioned with a ratio of 8:8:3:2 for the first source domain, the second source domain, the target domain (training), and the target domain (test), respectively. Each experiment was repeated 10 times. The statistical results with means and standard deviations are reported in Table 7. For single source based transfer learning methods, the results that were obtained based on different sources have been reported. The results in Table 7 show that although more promising results were obtained with the EKL-TSK-FS, a further improvement in performance can be achieved with its extended version, the MS-EKL-TSK-FS, or a competitive performance can at least be realized in comparison to that obtained by the single source based method when the best source domain is adopted.

Table 7 Generalization performance (J) of the EKL-TSK-FS and the MS-EKL-TSK-FS in the multi-source fermentation process modeling dataset

Dataset	Source	EKL-TSK-FS (D2+Knowledge)	MS-EKL-TSK-FS (D2+Knowledge)
$S(h+2)$	D1-1	0.1241(0.0226)*	**0.1169(0.0109)**
	D1-2	0.1260(0.0149)	
$X(h+2)$	D1-1	0.3916(0.0356)	**0.3090(0.0174)**
	D1-2	0.3605(0.0244)	
$P(h+2)$	D1-1	0.1485(0.0213)	**0.1397(0.0157)**
	D1-2	0.1579(0.0160)	

* In a(b), a and b denote the mean and standard deviation, respectively.

7. CONCLUSIONS

In this study, an enhanced knowledge-leverage based TSK fuzzy system modeling method was proposed in order to overcome the weaknesses of the knowledge-leverage based TSK fuzzy system modeling method. Two enhanced knowledge-leverage strategies were introduced to improve the transfer learning abilities for the learning of the antecedent parameters and consequent parameters, respectively. With these enhanced knowledge-leverage learning abilities, the proposed method showed a better modeling effect than existing knowledge-leverage based TSK fuzzy modeling methods and other related methods on synthetic and real-world datasets. Furthermore, the proposed method was extended to the multi-source domain scene.

Despite the promising performance of the proposed method, there is still room for improvement. For example, more advanced transfer learning mechanisms are still needed for the development of more adaptive TSK fuzzy system modeling methods. The development of more advanced transfer learning mechanisms for other types of fuzzy systems such as ML-type fuzzy systems and type-2 fuzzy systems, is also very significant. Future work will focus on these issues.

REFERENCES

[1] L.X. Duan, D. Xu, I. W. Tsang, “Domain adaptation from multiple sources: a domain-dependent regularization approach,” *IEEE Trans. Neural Netw. Learning Syst., vol. 23*, no. 3, pp. 504-518, 2012.
[2] S.J. Pan and Q. Yang, “A survey on transfer learning,” *IEEE Trans. Knowledge Data Engineering,* vol. 22, no. 10, pp. 1345–1359, 2010.
[3] L.X. Duan, I.W. Tsang, D. Xu, “Domain Transfer Multiple Kernel Learning,” *IEEE Trans. Pattern Anal. Mach. Intell., vol. 34*, no.3, pp. 465-479, 2012.
[4] W.H. Jiang and F.L. Chung, “Transfer Spectral Clustering,” *Proc. European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD)*, Bristol, UK, 2012.
[5] Z.H. Deng, Y.Z. Jiang, F.L. Chung, H. Ishibuchi, S.T. Wang, "Knowledge-Leverage Based Fuzzy System and Its Modeling." *IEEE Trans. Fuzzy Systems*, vol. 21, no. 4, pp. 597-609, 2013.

[6] Z.H. Deng, Y.Z. Jiang, K.S. Choi, F.L. Chung, S.T. Wang, "Knowledge-Leverage-Based TSK Fuzzy System Modeling," *IEEE Trans. Neural Networks and Learning Systems*, vol. 24, no. 8, pp. 1200-1212, 2013.
[7] J.W. Tao, K.F.L. Chung, S.T. Wang, "On minimum distribution discrepancy support vector machine for domain adaptation." Pattern Recognition, vol. 45, no.11, pp. 3962-3984, 2012.
[8] X. Liao, Y. Xue, and L. Carin, "Logistic regression with an auxiliary data source," *Proc. 21st Int. Conf. Machine Learning*, pp. 505-512, 2005.
[9] J. Huang, A. Smola, A. Gretton, K.M. Borgwardt, and B. Schölkopf, "Correcting sample selection bias by unlabeled data," *Proc. 19th Ann. Conf. Neural Information Processing Systems*, 2007.
[10] S. Bickel, M. Brückner, and T. Scheffer, "Discriminative learning for differing training and test distributions," *Proc. 24th Int. Conf. Machine Learning*, pp. 81-88, 2007.
[11] M. Sugiyama, S. Nakajima, H. Kashima, P.V. Buenau, and M. Kawanabe, "Direct importance estimation with model selection and its application to covariate shift adaptation," *Proc. 20th Ann. Conf. Neural Information Processing Systems*, 2008.
[12] N.D. Lawrence and J.C. Platt, "Learning to learn with the informative vector machine," *Proc. 21st Int. Conf. Machine Learning*, 2004.
[13] A. Schwaighofer, V. Tresp, and K. Yu, "Learning Gaussian process kernels via hierarchical Bayes," *Proc. 17th Ann. Conf. Neural Information Processing Systems*, pp. 1209-1216, 2005.
[14] J. Gao, W. Fan, J. Jiang, and J. Han, "Knowledge transfer via multiple model local structure mapping," *Proc. 14th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, pp. 283-291, 2008.
[15] L. Mihalkova, T. Huynh, and R.J. Mooney, "Mapping and revising Markov logic networks for transfer learning," *Proc. 22nd Assoc. for the Advancement of Artificial Intelligence (AAAI) Conf. Artificial Intelligence*, pp. 608-614, 2007.
[16] L. Mihalkova and R.J. Mooney, "Transfer learning by mapping with minimal target data," *Proc. Assoc. for the Advancement of Artificial Intelligence (AAAI '08) Workshop Transfer Learning for Complex Tasks*, 2008.
[17] J. Davis and P. Domingos, "Deep transfer via second-order Markov logic," *Proc. Assoc. for the Advancement of Artificial Intelligence (AAAI '08) Workshop Transfer Learning for Complex Tasks*, 2008.
[18] 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.
[19] P. Yang, Q. Tan, and Y. Ding, "Bayesian task-level transfer learning for non-linear regression," *Proc. Int. Conf. on Computer Science and Software Engineering,* pp. 62-65, 2008.
[20] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," *Proc. 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2012, pp. 2066–2073.
[21] W. Dai, Q. Yang, G. Xue, and Y. Yu, "Self-taught clustering," *Proc. 25th Int. Conf. Machine Learning*, pp. 200-207, 2008.
[22] Z. Wang, Y. Song, and C. Zhang, "Transferred dimensionality reduction," *Proc. European Conf. Machine Learning and Knowledge Discovery in Databases (ECML/PKDD '08)*, pp. 550-565, 2008.
[23] B. Quanz, J. Huan, "Large margin transductive transfer learning," *In Proceedings of the 18th ACM conference on Information and knowledge management*, pp. 1327-1336, 2009.
[24] J. M. Mendel, *Uncertain rule-based fuzzy logic systems: Introduction and new directions*. Upper Saddle River, NJ: Prentice-Hall, 2001.
[25] J.S.R. Jang, C.T. Sun, and E. Mizutani, *Neuro-fuzzy and soft-computing*. Upper Saddle River, NJ: Prentice-Hall, 1997.
[26] Z.H. Deng, K.S. Choi, F.L. Chung, S.T. Wang, "Scalable TSK fuzzy modeling for very large datasets using minimal-enclosing-ball approximation," *IEEE Trans. Fuzzy Systems*, vol. 19, no. 2, pp. 210-226, 2011.
[27] Z.H. Deng, Y.Z. Jiang, L.B. Cao and S.T. Wang, "Knowledge-Leverage Based TSK Fuzzy System with Improved Knowledge Transfer," *In Proceedings of FUZZ-IEEE 2014*, 2014.
[28] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its application to modeling and control," *IEEE Trans. Systems Man and Cybernetics*, vol.15, no.1, pp. 116-132, 1985.
[29] E.H. Mamdani, "Application of fuzzy logic to approximate reasoning using linguistic synthesis," *IEEE Trans. Computer*, vol. C-26 (12), pp. 1182- 1191, 1977.
[30] M.F. Azeem, M. Hanmandlu, and N. Ahmad, "Generalization of adaptive neural-fuzzy inference systems," *IEEE Trans. Neural Networks*, vol. 11, no. 6, pp. 1332- 1346, 2000.
[31] J.C. Bezdek, J. Keller, and R. Krishnapuram, *Fuzzy models and algorithms for pattern recognition and image processing*. San Francisco: Kluwer Academic Publishers, 1999.
[32] J. Leski, "TSK-fuzzy modeling based on ε-insensitive learning," *IEEE Trans. Fuzzy Systems*, vol. 13, no.2, pp. 181-193, 2005.

[33] J.S.R. Jang, "ANFIS: adaptive-network-based fuzzy inference system," *IEEE Trans. Systems, Man and Cybernetics*, 23, no. 3, pp. 665-685, 1993.
[34] I.W. Tsang, J.T. Kwok, and J.M. Zurada, "Generalized core vector machines," *IEEE Trans. Neural Networks*, vol. 17, no. 5, pp. 1126- 1140, 2006.
[35] B. Quanz, J. Huan, "Large margin transductive transfer learning," *In Proceedings of the 18th ACM conference on Information and knowledge management*, pp. 1327-1336, 2009.
[36] R.E. Fan, P.H. Chen, C.J. Lin, "Working Set Selection Using Second Order Information for Training Support Vector Machines," *Journal of Machine Learning Research,* vol. 6, pp. 1889-1918, 2005.
[37] E. Lughofer and S. Kindermann, "SparseFIS: Data-driven learning of fuzzy systems with sparsity constraints," *IEEE Trans. Fuzzy Syst.*, vol. 18,no. 2, pp. 396–411, Apr. 2010.
[38] C.F. Juang, S.H. Chiu, and S.J. Shiu, "Fuzzy system learned through fuzzy clustering and support vector machine for human skin color segmentation," *IEEE Trans. Systems Man and Cybernetics*, vol. 37, no. 6, pp. 1077-1087, 2007.
[39] C.F. Juang and C.D. Hsieh, "TS-fuzzy system-based support vector regression," *Fuzzy Sets and Systems*, vol. 160, no. 17, pp. 2486-2504, 2009.
[40] W. Dai, Q. Yang, G. Xue, and Y. Yu, "Boosting for transfer learning," in Proc. 24th Int. Conf. Mach. Learning, Jun. 2007, pp. 193–200.
[41] F. Zhuang, X. Cheng, P. Luo, S. J. Pan and Q. He, "Supervised representation learning: transfer learning with deep autoencoders," In *Proceedings of the 24th International Conference on Artificial Intelligence*, 2015, pp. 4119-4125, AAAI Press.
[42] F. Zhuang, P. Luo, C. Du, Q. He, Z. Shi and H. Xiong, "Triplex transfer learning: exploiting both shared and distinct concepts for text classification," *IEEE Transactions on Cybernetics*, vol. 44, no. 7, pp. 1191-1203, 2014.
[43] M. Long, J. Wang, G. Ding, S. J. Pan, and P. S. Yu, "Adaptation regularization: A general framework for transfer learning," IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 5, pp. 1076-1089, 2014.
[44] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," In *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 1717-1724), IEEE.
[45] W. Zhang, R. Li, T. Zeng, Q. Sun, S. Kumar, J. Ye, and S. Ji, "Deep model based transfer and multi-task learning for biological image analysis," In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, pp. 1475-1484, ACM.
[46] W. Pan, Q. Yang, "Transfer learning in heterogeneous collaborative filtering domains," *Artificial intelligence*, vol. 197, pp. 39-55, 2013.
[47] M. Long, J. Wang, J. Sun, and P.S. Yu, P. S, "Domain Invariant Transfer Kernel Learning," IEEE Transactions on Knowledge & Data Engineering, vol. 27, no. 6, pp. 1519-1532, 2015.
[48] T. Tommasi, F. Orabona, and B. Caputo, "Learning categories from few examples with multi model knowledge transfer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 5, pp. 928-941, 2014.
[49] N. Patricia, B. Caputo, "Learning to learn, from transfer learning to domain adaptation: A unifying perspective," In *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 1442-1449, IEEE.
[50] L. Shao, F. Zhu and X. Li, "Transfer learning for visual categorization: A survey," IEEE Transactions on Neural Networks and Learning Systems, vol. 26, no. 5, pp. 1019-1034, 2015.
[51] J. Lu, V. Behbood, P. Hao, H. Zuo, S. Xue, and G. Zhang, "Transfer learning using computational intelligence: A survey," *Knowledge-Based Systems*, vol. 80, pp. 14-23, 2015.
[52] L.P. Jing, M.K. Ng, and Z.X. Huang, "An entropy weighting k-means algorithm for subspace clustering of high-dimensional sparse data," *IEEE Trans. Knowledge and Data Engineering*, vol. 19, no. 8, pp. 1026–1041, 2007.
[53] Z.H. Deng, K.S. Choi, F.L. Chung, and S.T. Wang, "Enhanced soft subspace clustering integrating within-cluster and between-cluster information," *Pattern Recognition*, vol. 43, no. 3, pp. 767-781, 2010.