

# Fuzzy Transfer Learning in Heterogeneous Space using Takagi-Sugeno Fuzzy Models

Hua Zuo, Guangquan Zhang, and Jie Lu

Centre for Artificial Intelligence

University of Technology Sydney, 15 Broadway NSW 2007, Australia

Hua.Zuo@uts.edu.au, Guangquan.Zhang@uts.edu.au, and Jie.Lu@uts.edu.au

**Abstract.** Transfer learning is gaining increasing attention due to its ability to leverage previously acquired knowledge (a source domain with a large amount of labeled data) to assist in completing a prediction task in a related domain (a target domain with little labeled data). Many transfer learning methods have been proposed, and especially the fuzzy transfer learning method, which is based on fuzzy systems, has been developed because of its capability to deal with the uncertainty. However, there is one issue with fuzzy transfer learning that has not yet resolved: the domain adaptation methods for regression tasks in heterogeneous space are still scarce, and the relation of features in two domains have not been explored to assist the construction of target model. In this work, we proposed a new fuzzy transfer learning method, which constructs the transformed mappings for the domain-independent and domain-dependent features, separately. The existing fuzzy rules of the source domain are transferred to the target domain through modifying the input space using the mappings, and the parameters of the mappings are optimized by the few labeled target data. The experiments on real-world datasets validate the effectiveness of the proposed method and discuss the impact of some important parameters to the performance of the constructed target model.

**Keywords:** Transfer Learning, Fuzzy Rules, Domain Adaptation, Machine Learning

## 1 Introduction

Transfer Learning [1], as a branch of machine learning, has gained growing attention due to its ability of transferring knowledge between the domains with different data distributions or features. Transfer learning addresses the problem of how to leverage the knowledge acquired previously (the source domain with a large amount of labeled data) to improve the efficiency and accuracy of learning in one domain (the target domain with few labeled data) that in some way relates to the original domain.

The techniques of transfer learning are applied mainly in two scenarios. In the first scenario, a well-performed model is built based on the historical data, but because of the rapidly changing environment, the existing model is outdated and cannot fit the new data [2]. In the second scenario, which usually happens in the new emerging

area, few labeled or no labeled data is available, but there exists a related domain where a lot of labeled data is accessible [3]. Some examples of successful applications for the transfer learning methods include: classifying the French files leveraging the already-categorized English documents [4], predicting the status of the Australian banks, fail or survive, using the data of banks in America [5], and detecting the location of a user based on the previously collected WiFi data [6].

Since transfer learning belongs to a branch of machine learning, many methods in transfer learning are developed based on different prediction models, such as SVM [7], and neural networks [8]. There have been many survey papers that summarize the techniques in transfer learning, especially in some specific areas, such as activity cognition [9], reinforcement learning [10], collaborative recommendation [11], and computational intelligence [12].

The methods of fuzzy transfer learning have been proposed to deal with the phenomenon of uncertainty in transfer learning problems. The integration of fuzzy logic with transfer learning has drawn considerable attention in the literatures. The researchers have applied fuzzy sets to represent linguistic variables when feature values cannot be precisely described in numerical values, and to describe fuzzy distance for the retrieval of similar cases. Transferring implicit and explicit knowledge from similar domains is hidden and uncertain by nature, thus using fuzzy logic and fuzzy rule theory to handle the associated vagueness and uncertainty is apt and can improve transfer accuracy. Thus, many scholars have turned to fuzzy systems as a solution for transfer learning problems with promising results. Deng et al. [13] proposed a series of transfer learning methods, using a Takagi-Sugeno-Kang (TSK) fuzzy model, and developed novel fuzzy logic systems algorithms by defining two new objective functions. Further, their methods were applied to deal with the insufficient scenarios, for example, recognizing electroencephalogram signals in environments with a data shortage. Behbood et al. [14] proposed a fuzzy-based transfer learning approach to long-term bank failure prediction models with source and target domains that have different data distributions. Liu et al. [15] focus on the unsupervised heterogeneous domain adaptation problem, and presented a novel transfer learning model via  $n$ -dimensional fuzzy geometry and fuzzy equivalence relations. A metric based on  $n$ -dimensional fuzzy geometry is defined to measure the features' similarity of a domain. Then the shared fuzzy equivalence relations are proposed and used to make the numbers of clustering categories the same under the same value of  $\alpha$ , and knowledge can be transferred from the source domain to the target domain on heterogeneous space through the clustering categories.

Some work has been done to develop the domain adaptation ability of fuzzy rule-based models in regression tasks [16]. A set of algorithms are proposed for two different scenarios, where the datasets from the source domain and target domain are in homogeneous [17] and heterogeneous space [18], separately. In this paper, based on these works, we will explore the relation of features in two domains and apply it to assist the knowledge transfer process. The contribution of this work is using the relation of the features between domains to assist the construction of target model.

The reminder of this paper is structured as followed. Section 2 presents some definitions of transfer learning, and the Takagi-Sugeno fuzzy model. Section 3 details the

procedures of the proposed domain adaptation method in heterogeneous space. Experiments in Section 4 validate the effectiveness of the presented transfer learning method and discuss the impact of some important parameters in the model. The final section concludes the paper and outlines further work.

## 2 Preliminaries

In this section, some important definitions in transfer learning, and the Takagi-Sugeno fuzzy model are introduced to make the readers have a clear cognition of the background knowledge of this work.

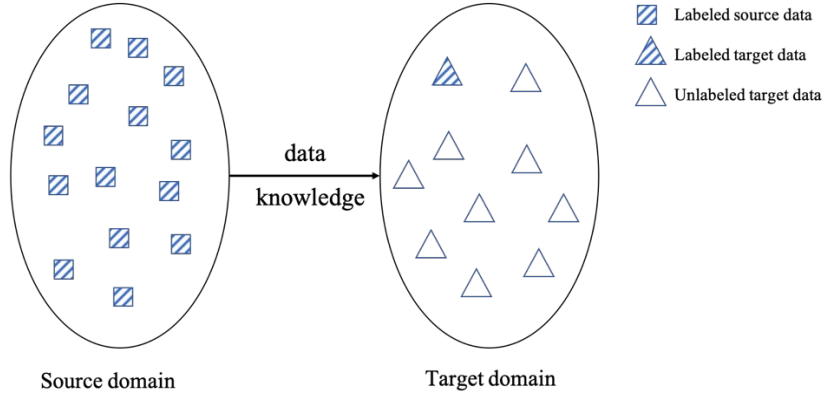
### 2.1 Definitions of Transfer Learning

A domain [1] is denoted by  $D = \{F, P(X)\}$ , where  $F$  is a feature space, and  $P(X)$ ,  $X = \{x_1, \dots, x_n\}$ , is the probability distribution of the instances.

A task [1] is denoted by  $T = \{Y, f(\cdot)\}$ , where  $Y \in R$  is the label, and  $f(\cdot)$  is an objective predictive function.

Definition 1 (Transfer Learning) [1]: Given a source domain  $D_s$ , a learning task  $T_s$ , a target domain  $D_t$ , and a learning task  $T_t$ , transfer learning aims to improve learning of the target predictive function  $f_t(\cdot)$  in  $D_t$  using the knowledge in  $D_s$  and  $T_s$  where  $D_s \neq D_t$  or  $T_s \neq T_t$ .

The process of transfer learning is illustrated in Fig.1.



**Fig. 1.** The process of transfer learning

In Brief, transfer learning aims to utilize the previously acquired knowledge (or data) in the source domain to support the prediction model's construction in the target domain.

## 2.2 Takagi-Sugeno Fuzzy Model

The prediction model is a commonly used regression model in the fuzzy system area, the Takagi-Sugeno (TS) fuzzy model, which is consist of fuzzy rules in a nonlinear way [19]. A Takagi-Sugeno fuzzy model, which contains  $c$  fuzzy rules, is represented as:

$$\text{If } \mathbf{x} \text{ is } A_i(\mathbf{x}, \mathbf{v}_i), \text{ then } y \text{ is } L_i(\mathbf{x}, \mathbf{a}_i) \quad i = 1, \dots, c \quad (1)$$

The TS fuzzy model could also be rewritten in the form of a neural network with the structure in Fig. 2. The first layer represents the input data, each neuron in the second layer represents a cluster, which also represents the condition of a fuzzy rule, and the third layer is the corresponding consequences of the fuzzy rules.

The construction of the TS model, a set of fuzzy rules, is based on a labeled dataset  $\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_N, y_N)\}$  using two procedures. In the first procedure, fuzzy C-means (FCM) [20] is applied to divide the data in an unsupervised learning process, so that clusters are learned, and the centers of the clusters are obtained. After getting the clusters, the coefficients of the linear functions, which are defined in each cluster, are calculated using the labeled datasets.

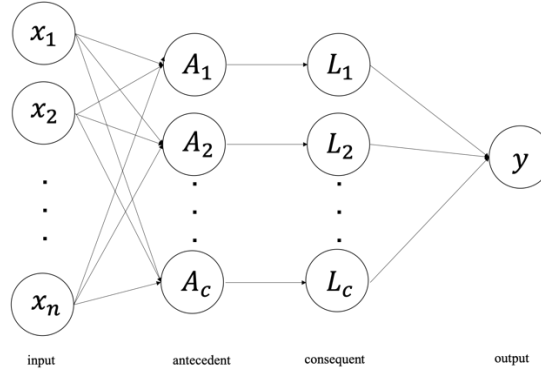


Fig. 2. TS model with the neural network structure

## 3 Knowledge Transfer between Domains in Heterogeneous Space

This section presents the method of transferring fuzzy rules from the source domain to a target domain in heterogeneous space. In order to facilitate elaborating the proposed method, the heterogeneous domain adaptation problem is stated with formula, and the description of the used variables and datasets is given.

### 3.1 Problem Statement

Consider there are one source domain with a large amount of labeled data, and a target domain with very few labeled data. Suppose the dataset in the source domain is  $\mathcal{S}$  and denoted as:

$$\mathcal{S} = \{(\mathbf{x}_1^s, y_1^s), \dots, (\mathbf{x}_{N_s}^s, y_{N_s}^s)\} \quad (2)$$

where  $(\mathbf{x}_k^s, y_k^s)$  is the  $k$ th input-output data pair in the source domain.  $\mathbf{x}_k^s \in R^n$  is an input variable with  $n$ -dimension, the label  $y_k^s \in R$  is the corresponding output, a continuous variable in a regression task, and  $N_s$  indicates the number of labeled data pairs in the source domain.

The target domain contains two subsets:  $\mathcal{T}_U$ , the one with labels, and  $\mathcal{T}_L$ , the one without labels:

$$\mathcal{T} = \{\mathcal{T}_L, \mathcal{T}_U\} = \{(\mathbf{x}_1^t, y_1^t), \dots, (\mathbf{x}_{N_{t1}}^t, y_{N_{t1}}^t), \dots, (\mathbf{x}_{N_t}^t, y_{N_t}^t)\} \quad (3)$$

where  $\{(\mathbf{x}_1^t, y_1^t), \dots, (\mathbf{x}_{N_{t1}}^t, y_{N_{t1}}^t)\}$  are the labeled data pairs in  $\mathcal{T}_L$ , and  $\{(\mathbf{x}_{N_{t1}+1}^t, \dots, \mathbf{x}_{N_t}^t)\}$  are the unlabeled data in  $\mathcal{T}_U$ . The numbers of instances in  $\mathcal{T}_L$  and  $\mathcal{T}_U$  are  $N_{t1}$  and  $N_t - N_{t1}$  respectively, and satisfy  $N_{t1} \ll N_t, N_{t1} \ll N_s$ .

Different with the homogeneous situation, in the heterogeneous domain adaptation problem, the feature space in the source and target domain are different. Here, we consider a special case, where dimensions of the feature space in two domains are the same, but the meanings of the features are not identical.

Suppose the feature spaces in the source and target domains are  $F_s$  and  $F_t$ :

$$F_s = (x_1^c, x_2^c, x_1^s, \dots, x_{n-2}^s) \quad (4)$$

$$F_t = (x_1^c, x_2^c, x_1^t, \dots, x_{n-2}^t) \quad (5)$$

where  $x_1^c$  and  $x_2^c$  are the two common features, which can be regarded as the domain-independent features, and the distributions of them are not identical in both source and target domains.  $(x_1^s, \dots, x_{n-2}^s)$ ,  $\dots$ , and  $(x_1^t, \dots, x_{n-2}^t)$  are the domain-dependent features, which own different meaning in two domains. Here, we set the number of domain-independent features as two as an example, and it could be any number that doesn't exceed the dimension of the feature space.

Since the number of labeled data is sufficient in  $\mathcal{S}$ , a well-performed model could be built for the source domain. Due to the different meanings of the features, however, the models for the source domain cannot be used directly to solve the regression tasks in the target domain. But the common features in the feature spaces of two domains provide the bridge that could transfer the shared knowledge between domains.

### 3.2 Knowledge Transfer in Heterogeneous Space across Domains

The method of transferring knowledge from the source domain to the target domain in heterogeneous space could be summarized into three steps:

Step 1: Construct a TS model for the source domain.

Based on the labeled data in  $\mathcal{S}$ , a TS fuzzy model  $M^s$  is built, and a set of fuzzy rules are obtained.

$$\text{If } \mathbf{x}_k^s \text{ is } A_i(\mathbf{x}_k^s, \mathbf{v}_i^s), \text{ then } y_k^s \text{ is } L_i(\mathbf{x}_k^s, \mathbf{a}_i^s) \quad i = 1, \dots, cs \quad (6)$$

Each rule, actually, is represented by the centers of the clusters  $\mathbf{v}_i^s$  and the coefficients of the linear functions  $\mathbf{a}_i^s$ . After the construction of the model  $M^s$ , a set of  $\mathbf{v}_i^s$  and  $\mathbf{a}_i^s$  are obtained, and the source data won't be used anymore, which could keep the privacy of the data, especially in some sensitive areas, for example the medical data.

The fuzzy rules in (6) have a high prediction accuracy on source data  $\mathcal{S}$ , but a poor performance on target data  $\mathcal{T}$ .

Step 2: Modify the existing rules in  $M^s$  to fit the target data.

The input space of the target data is changed through mappings to make the modified rules become compatible with target data. Since there are two types of features, domain-independent and domain-dependent, two different strategies are applied to change the features in the input space. The structures of mappings are shown in Fig. 3.

For the domain-independent features, which are shared between the source and target domains, each input variable is changed using a three layers network. Although these features have the same meanings between domains, their distributions are quite different. Each domain-independent variable is assumed to be governed by some hidden features, so the different hidden features or the different weights of the features will lead to the difference of the distributions in two domains. Therefore, the idea of applying the mapping in Fig. 3 aims to adjust the hidden features in number or weight so that the modified distribution of the domain-independent features could fit the target data.

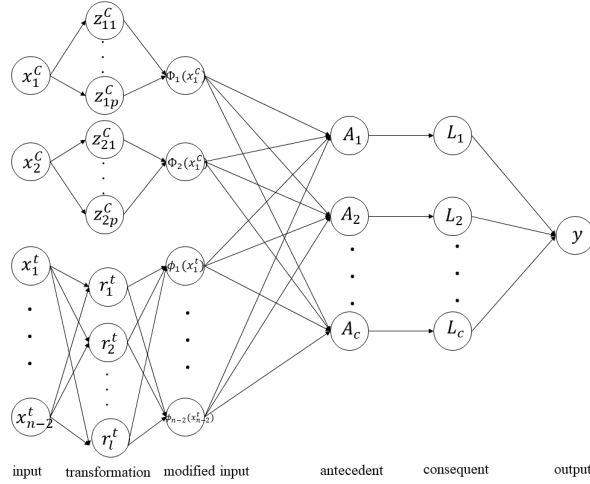


Fig. 3. Modification of input space for heterogeneous domain adaptation

For the domain-dependent features, since there is no clear relation between the features in two domains, a method of fully connection with a hidden is applied to change the distributions of the domain-dependent features.

Step 3: Optimize the parameters of mappings using target data

The parameters in the mappings are obtained through a supervised learning way. The labeled target data set  $\mathbf{T}_L$  is fed to the model and optimize the model to fit the target data. Denote  $(\Phi(x_1^c, x_2^c), \phi(x_1^t, \dots, x_{n-2}^t)) \equiv \Psi(\mathbf{x}^t)$ .

The following objective function is minimized:

$$Q = \sqrt{\frac{1}{N_{t1}} \sum_{k=1}^{N_{t1}} (\sum_{i=1}^{CS} A_i(\Psi(\mathbf{x}_k^t), \Psi(\mathbf{v}_i^s) L_i(\Psi(\mathbf{x}_k^t), \mathbf{a}_i^s) - y_k^t)^2} + \frac{\lambda}{2} \mathbf{w}^T \mathbf{w} \quad (7)$$

The first term in (7) is the approximation error that aims to minimize the gap between the output of the modified model and the target data's real output. The second term introduces a structural risk term into the objective function. The parameter  $\lambda$  indicates the tradeoff between the quality of an approximation and the complexity of the approximation function;  $\mathbf{w}$  is the vector of all the parameters optimized.

## 4 Experiments

The experiments using real-world datasets are implemented to validate the effectiveness of the proposed method in dealing with heterogeneous domain adaptation problem, and explore the impact of some important parameters in the transferring model.

Since the studies on regression problems of domain adaptation are scarce, especially in the heterogeneous space, there is no public datasets in these scenarios. In the work, therefore, two datasets from UCI Machine Learning Repository are used and modified to simulate the heterogeneous domain adaptation problems. A detailed description of modifying the datasets is provided to illustrate the datasets clearly. Further, all the models' construction apply the five-fold cross validation, the results are shown in the form of "mean $\pm$ variance". All the models are tested on the unlabeled target data to compare the ability of solving the regression tasks in target domain.

The first dataset concerns "Airfoil self-noise dataset". The dataset was split based on the "frequency" value. Five attributes, "frequency", "angle of attack", "chord length", "free-stream velocity", and "suction side displacement thickness", are used to predict the "scaled sound pressure level". Data with a "frequency" of greater than 800 hertz formed the source domain, with 1000 instances, the remaining data, 450 instances, was used for the target domain. Further, the attributes "chord length", "free-stream velocity", and "suction side displacement thickness" in the source domain were perturbed with random numbers to simulate the domain-dependent features. All the instances in the source domain are labeled, but only 10 instances in the target domain are labeled.

The number of clusters, i.e. the amount of the fuzzy rules, is a crucial parameter that determines the structure of the model. Although some techniques, for example the infinite Gaussian mixture model and the heuristic algorithm, have been applied to

explore the number of clusters, it is still not easy to determine in the high-dimension datasets. Therefore, the number of clusters is set as a hyper-parameter in the following experiments to explore the impact of it to the performance of transfer learning in fuzzy models.

With different number of clusters, the heterogeneous domain adaptation method is implemented, and the results are shown in Table 1. The first column in Table 1 indicates the number of clusters applied in the experiment. The results shown in the second to the fourth columns are the accuracies of three models on the unlabeled data in the target domain. The three models are: the source model built using source data, the target model built using only labeled target data, and the target data constructed using our proposed method. The results with the best performance are in bold.

**Table 1.** Results with different values of  $c$  in “Airfoil self-noise dataset”

Number of clusters	Performance of models on unlabeled target data		
	Source model	Target model using target data	Model using new method
2	0.5407 $\pm$ 0.0002	0.4279 $\pm$ 0.0088	<b>0.2013<math>\pm</math>0.0010</b>
3	0.5524 $\pm$ 0.0001	0.2805 $\pm$ 0.0009	<b>0.1921<math>\pm</math>0.0009</b>
4	0.5682 $\pm$ 0.0001	0.2550 $\pm$ 0.0052	<b>0.2184<math>\pm</math>0.0007</b>
5	0.5672 $\pm$ 0.0001	0.2260 $\pm$ 0.0015	<b>0.1721<math>\pm</math>0.0001</b>
6	0.5744 $\pm$ 0.0001	0.2175 $\pm$ 0.0008	<b>0.2165<math>\pm</math>0.0007</b>
7	0.5806 $\pm$ 0.0001	0.2067 $\pm$ 0.0009	<b>0.1802<math>\pm</math>0.0011</b>

Analyzing the results in Table 1, the high values in the first column indicate that the model built for the source domain does not fit to the target data, and the not small values in the third column manifest the poor performance of the target model only using the labeled target data, which also verify that the labeled target data is not sufficient to construct a well prediction model. Additionally, comparing the results in the second, third, and fourth columns, we can see that in each experiment with different number of clusters, the accuracies of the model applied the presented new method is higher than the accuracies of the other models, which indicates that our method is superior than other methods.

In the presented method, the number of neurons used to construct the mappings is an important parameter that determines the performance of the constructed target model. Since the input features are changed through two different ways, two parameters,  $p$  and  $q$ , are discussed here to explore the impact of them to the performance of the constructed model. The target model using the new method is built with different values of  $p$  and  $q$ . For comparison, the performance of the source model and target model using only labeled target data are shown in Table 2.

The performance of the target model using the proposed method is shown in Table 3. Both the values of  $p$  and  $q$  are changing from two to six, so twenty-five experiments are run to compare the results.



**Table 2.** Results of two compared models

Source model	$0.5684 \pm 0.0001$
Target model using target data	$0.2463 \pm 0.0019$

**Table 3.** Results with different values of p and q

$\begin{matrix} q \\ p \end{matrix}$	2	3	4	5	6
2	$0.1889 \pm 0.0003$	$0.1984 \pm 0.0001$	$0.2004 \pm 0.0009$	$0.2035 \pm 0.0004$	$0.2164 \pm 0.0013$
3	$0.1812 \pm 0.0001$	$0.1916 \pm 0.0006$	$0.2062 \pm 0.0017$	$0.2130 \pm 0.0012$	$0.2263 \pm 0.0023$
4	$0.1814 \pm 0.0005$	$0.1977 \pm 0.0009$	$0.2033 \pm 0.0013$	$0.2264 \pm 0.0031$	$0.2116 \pm 0.0002$
5	$0.1790 \pm 0.0003$	$0.2013 \pm 0.0011$	$0.2134 \pm 0.0007$	$0.1984 \pm 0.0008$	$0.2112 \pm 0.0019$
6	$0.1827 \pm 0.0002$	$0.1936 \pm 0.0008$	$0.1869 \pm 0.0009$	$0.1965 \pm 0.0009$	$0.2178 \pm 0.0017$

First, all the mean values in Table 3 is smaller than the mean values in Table 2, which validates the effectiveness of our method in deal with transfer learning in heterogeneous space. There is no obvious trend of the results with the change of p and q, and the best result appears when the p is equal to five and q is two.

In the second dataset, “Combined cycle power plant dataset (CCPP)”, four attributes “temperature”, “ambient pressure”, “relative humidity”, and “exhaust vacuum” were used to predict the “net hourly electrical energy output. The dataset was split into two source domains and a target domain using the attribute “temperature”; instances with a temperature of not greater than 25 degree were treated as source data, with 5000 instances, the remainder 2000 instances fell into the target domain. Also, the features “relative humidity”, and “exhaust vacuum” in the source domain were perturbed with random numbers to simulate the domain-dependent features. All the instances in the source domain are labeled, but only 10 instances in the target domain are labeled.

Similarly, a set of experiments are designed to find out the impact of the changing number of clusters to the presented method. The results are shown in Table 4. In the first five experiments shown in Table 4, the performance of the model using the presented method is superior than the other two models. However, in the last experiment, where the number of clusters is equal to seven, the performance of the model applying only the target labeled target data is better than the proposed method, and the reason might derive from the quality of the labeled target data. Comparing the results in the third and fourth columns, although the new method shows better capability in dealing with the heterogeneous domain adaptation problem, it does not show a significant advantage. Although the labeled target data has small amount, they cover most of the clusters of target data, which lead to the good quality of the model using only target data.

**Table 4.** Results with different values of  $c$  in “CCPP”

Number of clusters	Performance of models on unlabeled target data		
	Source model	Target model using target data	Model using new method
2	0.5444 $\pm$ 0.0000	0.1021 $\pm$ 0.0006	<b>0.0720<math>\pm</math>0.0000</b>
3	0.5422 $\pm$ 0.0000	0.0854 $\pm$ 0.0003	<b>0.0738<math>\pm</math>0.0001</b>
4	0.5483 $\pm$ 0.0000	0.0857 $\pm$ 0.0001	<b>0.0728<math>\pm</math>0.0000</b>
5	0.5511 $\pm$ 0.0000	0.0868 $\pm$ 0.0001	<b>0.0686<math>\pm</math>0.0000</b>
6	0.5487 $\pm$ 0.0000	0.0817 $\pm$ 0.0000	<b>0.0803<math>\pm</math>0.0002</b>
7	0.5492 $\pm$ 0.0000	<b>0.0815<math>\pm</math>0.0000</b>	0.0872 $\pm$ 0.0002

The experimental results with varying values of  $p$  and  $q$  are shown in Tables 5 and 6. In the twenty-five experiments, the number of clusters is set to five, and the values of  $p$  and  $q$  go through the integers between two and six. Similarly, the proposed method shows its superiority in these experiments when comparing with the source model and the model built using target data only. The best result appears when the values of  $p$  and  $q$  are equal to six and four, separately. Additionally, the low variance of the results in all the experiments show the good generalization ability of the model constructed using the presented method.

**Table 5.** Results of two compared models

Source model	0.5492 $\pm$ 0.0001
Target model using target data	0.0803 $\pm$ 0.0001

**Table 6.** Results with different values of  $p$  and  $q$ 

$q \backslash p$	2	3	4	5	6
2	0.0735 $\pm$ 0.0001	0.0699 $\pm$ 0.0000	0.0717 $\pm$ 0.0000	0.0740 $\pm$ 0.0001	0.0750 $\pm$ 0.0000
3	0.0707 $\pm$ 0.0000	0.0717 $\pm$ 0.0001	0.0761 $\pm$ 0.0001	0.0746 $\pm$ 0.0000	0.0707 $\pm$ 0.0000
4	0.0706 $\pm$ 0.0000	0.0710 $\pm$ 0.0000	0.0718 $\pm$ 0.0000	0.0766 $\pm$ 0.0002	0.0720 $\pm$ 0.0001
5	0.0677 $\pm$ 0.0000	0.0695 $\pm$ 0.0000	0.0680 $\pm$ 0.0000	0.0759 $\pm$ 0.0000	0.0694 $\pm$ 0.0000
6	0.0687 $\pm$ 0.0001	0.0693 $\pm$ 0.0000	0.0674 $\pm$ 0.0000	0.0710 $\pm$ 0.0000	0.0741 $\pm$ 0.0000

## 5 Conclusion and Further Study

This work explores the transfer learning problems in heterogeneous space. The features in the source and target domains are identified by domain-independent features,

which are shared between domains, and domain-dependent features, which only exist in an individual domain. Mappings with different structures are constructed for the two types of features, separately. For the domain-independent features, a shallow network is built for each feature to change the input distribution, and for the domain-dependent features, all the domain-dependent features are used to construct the transformation mapping. The parameters of the mappings are obtained through an optimization process based on the labeled target data. The experiments show that the accuracy of the model using our method is superior than the source model, and the model built with insufficient target labeled data. And an appropriate number of clusters is important to the performance of the target model. The change of values of  $p$  and  $q$  has slight impact to the final results.

The method presented in this paper focuses on the situation, where the dimensions of the feature space in two domains are the same, a special case in heterogeneous transfer learning. More general and challenging case, that the dimensions of the input data in two domains are different, will be considered and discussed in the future studies.

### Acknowledgement

This work was supported by Australian Research Council under DP 170101623.

### References

1. S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345-1359, 2010.
2. B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 2066-2073.
3. 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 Systems*, vol. 23, no. 6, pp. 1917-1935, 2015.
4. M. Xiao and Y. Guo, "Feature space independent semi-supervised domain adaptation via kernel matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 1, pp. 54-66, 2015.
5. V. Behbood, J. Lu, and G. Zhang, "Fuzzy bridged refinement domain adaptation: Long-term bank failure prediction," *International Journal of Computational Intelligence and Applications*, vol. 12, no. 01, 2013.
6. S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE Transactions on Neural Networks*, vol. 22, no. 2, pp. 199-210, 2011.
7. 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.

8. S. Hoo-Chang *et al.*, "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, p. 1285-1298, 2016.
9. D. Cook, K. D. Feuz, and N. C. Krishnan, "Transfer learning for activity recognition: A survey," *Knowledge and Information Systems*, vol. 36, no. 3, pp. 537-556, 2013.
10. M. E. Taylor and P. Stone, "Transfer learning for reinforcement learning domains: A survey," *Journal of Machine Learning Research*, vol. 10, no. Jul, pp. 1633-1685, 2009.
11. W. Pan, "A survey of transfer learning for collaborative recommendation with auxiliary data," *Neurocomputing*, vol. 177, pp. 447-453, 2016.
12. 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.
13. 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 Transactions on Intelligent Systems and Technology (TIST)*, vol. 8, no. 1, 2016.
14. V. Behbood, J. Lu, and G. Zhang, "Fuzzy refinement domain adaptation for long term prediction in banking ecosystem," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1637-1646, 2014.
15. F. Liu, G. Zhang, and J. Lu, "Unconstrained fuzzy feature fusion for heterogeneous unsupervised domain adaptation," in *2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pp. 1-8, 2018.
16. H. Zuo, G. Zhang, W. Pedrycz, V. Behbood, and J. Lu, "Granular fuzzy regression domain adaptation in Takagi-Sugeno fuzzy models," *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 2, pp. 847-858, 2018.
17. H. Zuo, G. Zhang, W. Pedrycz, V. Behbood, and J. Lu, "Fuzzy regression transfer learning in Takagi-Sugeno fuzzy models," *IEEE Transactions on Fuzzy Systems*, vol. 25, no. 6, pp. 1795-1807, 2017.
18. H. Zuo, J. Lu, G. Zhang, and W. Pedrycz, "Fuzzy rule-based domain adaptation in homogeneous and heterogeneous spaces," *IEEE Transactions on Fuzzy Systems*, vol. 27, no. 2, pp. 348-361, 2018.
19. M. L. Hadjili and V. Wertz, "Takagi-Sugeno fuzzy modeling incorporating input variables selection," *IEEE Transactions on Fuzzy Systems*, vol. 10, no. 6, pp. 728-742, 2002.
20. J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: The fuzzy c-means clustering algorithm," *Computers & Geosciences*, vol. 10, no. 2-3, pp. 191-203, 1984.