

Fuzzy Domain Adaptation Using Unlabeled Target Data

Hua Zuo, Guangquan Zhang, and Jie Lu

DeSI Lab, Centre for Artificial Intelligence, FEIT, University of Technology Sydney,
81 Broadway, Ultimo, NSW, 2007, Australia
{hua.zuo, guangquan.zhang, jie.lu}@uts.edu.au

Abstract. Transfer learning has been emerging recently and gaining more attention because of its ability to deal with “small labeled data” issue in new markets and for new products. It addresses the problem of leveraging knowledge acquired from previous domain (a source domain with a large amount of labeled data) to improve the accuracy of tasks in the current domain (a target domain with little labeled data). Fuzzy rule-based transfer learning methods are developed due to the ability to dealing with the uncertainty in domain adaptation scenarios. Although some effort is made to develop the fuzzy methods, they only apply the knowledge of the labeled data in the target domain to assist the model’s construction. This work develops a new method that explores and utilizes the information contained in the unlabeled target data to improve the performance of the new constructed model. The experiments on both synthetic datasets and real-world datasets illustrate the effectiveness of our method, and also give the application scope of applying it.

Keywords: Domain adaptation, Transfer learning, Machine learning, Fuzzy rules, Regression

1 Introduction

Machine learning [1] has gained a great achievement in many areas, such as finance, military, entertainment, and so on. And many machine learning methods are developed to handle the practical situations [2]. Although these methods work well in some cases, there is a big obstacle that impeded the further development of the traditional machine learning methods. This obstacle comes from an assumption that the model only works well in the condition that the training data and testing data have the same statistical characteristics, i.e. the same feature space and distributions. But in the data-shortage and rapid-changing environments, the constructed model always has a poor performance, and building a new is impossible due to the insufficient labeled data.

Human own the ability of applying the knowledge acquired previously to solve the current task. For example, recognizing an apple will be helpful for identifying a new fruit, such as a peach or a pear. Because of this ability, human could continuously accumulated knowledge and adapt to the new and challenging environment. Transfer learning has been emerging recently and becoming more and more popular due to its ability of knowledge transfer.

There are increased attention focusing on transfer learning [3], and more methods are developed to handle the real cases in artificial intelligence to expand the application of transfer learning. Some well-known examples of transfer learning include prediction, image recognition, recommend systems, and natural language processing. Since the development of transfer learning is based on machine learning, many notable machine learning models [6-8] are applied as the basic learning in the transfer learning methods. Additionally, researchers in deep learning exploring the transfer ability of deep models [9]. For more information about transfer learning, there are some well-written survey papers that summarize the current transfer learning methods and give the clear categories to review them [10, 11].

Although transfer learning exhibits an upward trend, there is still a huge gap between existing work and domain adaptation tasks. For instance, most of the current transfer learning focus on solving the classification problems, but there is little work on the regression tasks in the domain adaptation problems. Additionally, the ignorance of the uncertainty phenomenon in the transfer learning problems weak the application scope of the current works. Since only little labeled or no labeled data available in the target domain, the insufficient information lead to the uncertainty in the learning and model's construction process. However, the application of fuzzy systems in the transfer learning problems has shown a good results and light the way of handling the uncertainty issues.

We have done some work at solving the domain adaptation problems in regression tasks using fuzzy rule-based models [12, 13]. We have proposed three algorithms that deal with three different fuzzy transfer learning cases separately. In the first case, we consider the discrepancy of the distributions in the feature space, which will lead to the different conditions of fuzzy rules in the source and target domain. An algorithm of changing the input space through mappings is presented to solve the distribution gap between two domains. In the second case, other than the feature distributions, consider the difference in the output space, and an algorithm of changing the linear functions is proposed to adjusting the output space to make it fitted with the target data. The third algorithm combines the first two, modifies both the conditions and conclusion of the fuzzy rules to make them compatible with the target domain. All the work in these papers use the labeled target data to lead the construction of transformation mappings, and ignore the data without labels. Here, we propose a new method that explores and utilizes the knowledge contained in the unlabeled target data to improve the performance of the constructed target model.

The structure of this work is as follows. Section 2 presents some basic definitions in transfer learning, and the learning model applied in our method, Takagi-Sugeno fuzzy model. In Section 3, we propose a new method, which uses both labeled and unlabeled target data. In section 4, synthetic and real-world datasets are used to analyse the performance of our method and test its effectiveness in dealing with practical situations. The final section concludes the paper and outlines future work.

2 Preliminary

Some definitions are introduced first to give the readers the basic knowledge of transfer learning. And then, a fuzzy rule-based system, Takagi-Sugeno fuzzy model, is formulated.

2.1 Transfer Learning

Definition 1 (Domain) [3]: A domain is denoted by $D = \{F, P(X)\}$, where F is a feature space, and $P(X)$, $X = \{x_1, \dots, x_n\}$, are the probability distributions of the instances.

Definition 2 (Task) [3]: A task is denoted by $T = \{Y, f(\cdot)\}$, where $Y \in R$ is the output, and $f(\cdot)$ is an objective predictive function.

Definition 3 (Transfer Learning) [3]: 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 $T_s \neq T_t$.

Transfer learning uses the knowledge obtained from previous domains (source domain) to help build the model for dealing with the tasks in the current domain (target domain).

2.2 Fuzzy Rule-based Model

The basic learning model used here is the Takagi-Sugeno fuzzy model, which consists of c rules as follows:

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

where \mathbf{x} is the input, and y_i is the output of applying the corresponding rule. \mathbf{v}_i is the centers of the prototype (cluster), and \mathbf{a}_i determines the linear functions in the conclusions of the fuzzy rules. Thus, the output of the fuzzy system is y with the following representation.

$$y = \sum_{i=1}^c A_i(\mathbf{x}, \mathbf{v}_i) L_i(\mathbf{x}, \mathbf{a}_i) \quad (2)$$

The construction of the Takagi-Sugeno fuzzy model involves a learning process based on a given labeled datasets. First, the data are divided based on Fuzzy C-means algorithm, which could help cluster the data and find out the centers of the clusters \mathbf{v}_i . Based on the \mathbf{v}_i , the coefficients of \mathbf{a}_i are computed through an optimization procedure.

3 Methodology

In our previous papers, we proposed the methods of changing the input and output spaces of the source domain to fit the current tasks. The labeled target data are used to guide the construction of the mappings that connects the domains. But the unlabeled target data are not used to help the construction of target model. In the transfer learning

scenarios, there are a large amount of target data without labels that also contains much information of target domain. Therefore, how to utilize the unlabeled target data is a critical step in enhancing the performance of transfer learning between domains.

In this work, the knowledge contained in the unlabeled target data will be explored and applied to improve the performance of the constructed model for the target domain. The following steps outline this fuzzy rule-based domain adaptation method, which utilizes target data with and without labels for solving the regression tasks in target domain.

Step 1: Train a fuzzy model (fuzzy rules) for the source domain.

In the source domain, a mass of labeled data are available. Suppose the dataset in the source domain is denoted as $\mathbf{D} = \{(\mathbf{x}_1^s, y_1^s), \dots, (\mathbf{x}_{N_s}^s, y_{N_s}^s)\}$, where $\mathbf{x}_k^s \in R^n$ ($k = 1, \dots, N_s$) is an n -dimensional variable, $y_k^s \in R$ is a continuous variable, and N_s gives the number of labeled data. Based on the dataset \mathbf{D} , a supervised learning process is executed to train the source model and gain a set of fuzzy rules.

Referring to the numbers of fuzzy rules in the source and target domains, we want to claim that the rules of source domain could be modified and transferred to solve the target tasks as if the number of fuzzy rules in the source domain is greater than in the target domain. Since the Takagi-Sugeno fuzzy model uses nonlinearly weighted linear functions to fit a curve. Each cluster indicates a separate area in the input space, and the corresponding linear function represents the action applied in that area. More clusters, or fuzzy rules, represents more precise of the partition and action described in the output space. Thus, it is reasonable to set an adequate number of fuzzy rules when building a Takagi-Sugeno fuzzy model to get good performance.

We consider two cases here to indicate the relationship of the numbers of fuzzy rules in two domains. If the source domain has no less fuzzy rules' number than the target domain, then the fuzzy rules of source domain could be modified and used to handle the regression tasks in the target domain. If the source domain has less fuzzy rules' number than the target domain, we could adopt the strategy of retaining the source model with rules no less than that in the target domain. Therefore, this also can be regarded as a criteria of selecting an appropriate domain from multiple domains for the target domain.

Therefore, determining the fuzzy rules' number is quite important when building a Takagi-Sugeno fuzzy model. Although we claim that using more rules to construct a Takagi-Sugeno fuzzy model is reasonable, we still need the prior knowledge to estimate the number of fuzzy rules for a specific domain or dataset. Here, we apply the IGMM model [14] to find out the data's structure and provide a guide to determine the number of trained rules. IGMM implements the process of mixing Gaussian distributions to fit the data distribution, which detects the data structure in the data-based learning process.

After analysing the results from IGMM, a model M^s is trained based on the source dataset \mathbf{D} , and a set of fuzzy rules are obtained with formulation as follows:

$$\text{if } \mathbf{x}_k^s \text{ is } A_i(\mathbf{x}_k^s, \mathbf{v}_i^s), \text{ then } y_k^s \text{ is } L_i(\mathbf{x}_k^s, \mathbf{a}_i^s) \quad i = 1, \dots, c \quad (3)$$

There are c fuzzy rules, and each rule is governed by the center of cluster \mathbf{v}_i^s , and the linear function \mathbf{a}_i^s .

Step 2: Modify the fuzzy rules of source domain to fit the target tasks.

The target dataset \mathbf{H} consists of two subsets: \mathbf{H}_L with labeled data, and \mathbf{H}_U with unlabeled data. $\mathbf{H} = \{\mathbf{H}_L, \mathbf{H}_U\} = \{(\mathbf{x}_1^t, y_1^t), \dots, (\mathbf{x}_{N_{t1}}^t, y_{N_{t1}}^t)\}, \{\mathbf{x}_{N_{t1}+1}^t, \dots, \mathbf{x}_{N_t}^t\}$, where $\mathbf{x}_k^t \in R^n$ ($k = 1, \dots, N_t$) is the n -dimensional input variable, $y_k^t \in R$ corresponds to the labels for the data in \mathbf{H}_L . The numbers of data in \mathbf{H}_L and \mathbf{H}_U are N_{t1} and $N_t - N_{t1}$ respectively, and satisfy $N_{t1} \ll N_t, N_{t1} \ll N_s$.

Since the distributions of $\{\mathbf{x}_1^s, \dots, \mathbf{x}_{N_s}^s\}$ and $\{\mathbf{x}_1^t, \dots, \mathbf{x}_{N_t}^t\}$ are different, the model trained on \mathbf{D} could not be used to do the prediction tasks in \mathbf{H} .

We adopt the approach of changing the input space by constructing a mapping between the input variables between two domains [13]. Different with our previous method, which only applies the labeled target data to train the mapping, this work uses the information contained in unlabeled target data to enhance the accuracy of the target model.

After the transformation of the mapping in the input space, the fuzzy rules in M^s are changed, and target model M^t is obtained:

$$\text{if } \mathbf{x}_k^t \text{ is } A_i(\Phi(\mathbf{x}_k^t), \Phi(\mathbf{v}_i^s)), \text{ then } y_k^t \text{ is } L_i((\Phi(\mathbf{x}_k^t), \mathbf{a}_i^s)) \quad i = 1, \dots, c \quad (4)$$

where Φ is the transformation mapping for the input space. $\Phi = [\Phi_1 \dots \Phi_n]$ indicates that the mapping for each input variable is built separately, and the structures of them are the same, which is constructed of network with one hidden layer. The detailed structure of the mappings could refer to our previous paper [13].

To optimize the parameters in Φ , target data with and without labels are used to train and modify the existing fuzzy rules. The optimized cost function is:

$$S = \sqrt{\frac{1}{N_{t1}} \sum_{k=1}^{N_{t1}} \left(\sum_{i=1}^c \frac{A_i(\Phi(\mathbf{x}_k^t), \Phi(\mathbf{v}_i^s))}{\sum_{j=1}^c A_j(\Phi(\mathbf{x}_k^t), \Phi(\mathbf{v}_j^s))} L_i(\Phi(\mathbf{x}_k^t), \mathbf{a}_i^s) - y_k^t \right)^2} + \lambda_1 \sqrt{\frac{1}{N_{t1} * h} \sum_{k=1}^{N_{t1}} \sum_{l=1}^h (y_k^t - y_k^t(l))^2 * \exp(-\|\mathbf{x}_k^t - \mathbf{x}_k^t(l)\|)} + \frac{\lambda_2}{2} w^T w \quad (5)$$

There are three items in the cost function in (5). The first item focuses on the labeled target data, which guides the learning process in a supervised way. The second item intend to utilize the unlabeled data to optimize the parameters of the mappings. In the regression problems, it is reasonable to assume that the closer data in the input space have similar outputs. With this assumption, the outputs of unlabeled target data are estimated and considered to be approximate to the output of the nearest instance with label. Thus, the h -nearest data $\{\mathbf{x}_k^t(1) \dots \mathbf{x}_k^t(h)\}$ in \mathbf{H}_U are found for each labeled target data \mathbf{x}_k^t , and the corresponding outputs for $\{\mathbf{x}_k^t(1) \dots \mathbf{x}_k^t(h)\}$ are expected to be similar with the output of \mathbf{x}_k^t . $\exp(-\|\mathbf{x}_k^t - \mathbf{x}_k^t(l)\|)$ defines the degree of the closeness to make sure that the closer data have more approximate outputs. The third item controls the complexity of the constructed prediction model.

4 Experiments

The synthetic datasets are applied first to validate our proposed method, and indicate the application scope of it. Secondly, our method is used to solve some real-world domain adaptation problems.

To evaluate this fuzzy method, the performance indexes are given in advance. The RMSE is chosen to test the model. In addition, the generalization ability of the constructed model is also important, so five-fold cross validation is used in the models' construction process. To keep consistence, when testing the performance of a model in solving the target tasks, the target dataset H_U is used to estimate the ability of the model in fitting target data.

4.1 Experiments on Synthetic Datasets

Three group of experiments are implemented using the synthetic datasets. We first compare our fuzzy method, the baselines, and some famous methods. The second and third groups of experiments are executed to find out the impact of the data's structure to the performance of the proposed method.

In each group of the experiments, three datasets with different numbers of clusters are generated, and each time, two of them are selected as a source domain, where all data are labeled, and a target domain, where only 1% data are labeled.

In the first group of experiments, the proposed method is compared with one baseline, the source model, and two famous methods in transfer learning, TCA and SA. There are three datasets: "3r", "4r" and "5r", and "3r" means this dataset is generated using three clusters. Since we have three datasets, six experiments are implemented. In Table 1, the datasets applied in each experiment are indicated in column one. For example, "5r to 4r" means the source and target datasets assigned in this experiment are "5r" and "4r", separately. The second to fifth columns show the RMSE of the four methods.

From Table 1, the performance of our method is better than the baseline, TCA, and SA, based on the smaller values of RMSE in the six experiments.

Table 1. Transferring results in different methods

Source to target	RMSE of models			
	baseline	TCA	SA	Our method
5r to 4r	5.23±0.00	7.88±0.00	7.58±0.00	0.68±0.01
5r to 3r	3.67±0.00	4.66±0.00	4.65±0.00	1.14±0.05
4r to 3r	0.97±0.00	2.19±0.00	2.37±0.00	0.65±0.00
3r to 4r	0.94±0.00	6.10±0.01	6.36±0.00	0.72±0.01
3r to 5r	4.16±0.00	3.25±0.00	3.21±0.00	1.57±0.00
4r to 5r	4.67±0.00	5.91±0.02	5.45±0.01	1.39±0.01

To validate the effectiveness of applying the target data without labels, the performances of the models constructed using or not using unlabeled target data H_U are compared. In addition, we consider two cases, where the structures of the data are different, to find out the impact of data's partition to the target model.

The boundaries of the clusters are very clear and ambiguous in the second and third groups of experiments. The results are shown in Tables 2 and 3, separately. Table 2 compares the performance the target models, which are built using or not using unlabeled target data. Similarly, Table 3 compares the RMSE of the constructed target model with and without H_U . Lower values are shown in bold.

Table 2. Target model built using/ not using H_U – second group

Source to target datasets	RMSE of the models	
	M^t (without H_U)	M^t (with H_U)
5r to 4r	1.0781±0.0004	1.0756±0.0004
5r to 3r	0.9352±0.0057	0.8962±0.0083
4r to 3r	2.1269±0.2059	2.0996±0.1718
3r to 4r	0.8876±0.0009	0.8457±0.0005
3r to 5r	2.5273±0.0007	2.5397±0.0014
4r to 5r	3.0755±0.0110	3.0614±0.0037

Table 3. Target model built using/ not using H_U – third group

Source to target datasets	RMSE of the models	
	M^t (without H_U)	M^t (with H_U)
5r to 4r	2.05±0.30	2.11±0.31
5r to 3r	2.45±0.71	2.69±1.01
4r to 3r	3.00±2.24	2.39±1.32
3r to 4r	1.01±0.00	1.05±0.00
3r to 5r	5.50±0.51	5.45±0.35
4r to 5r	4.79±0.25	4.51±0.62

From the results in Tables 2 and 3, we can see that if the partition of data in the input space has obvious clusters, the use of unlabeled data could enhance the model's accuracy notably. But if the division of data's clusters is not clear, the application of H_U is not always an advanced result. This is because when the boundaries of input data are ambiguous, the labeled target data may fall into the junctions of the clusters, and the utilizing of H_U , finding the h -nearest unlabeled target data for each labeled target data, will lead to a poor performance of the target model.

4.2 Experiments on Real-world Datasets

Since most studies on transfer learning focus on classification problems, there are no publicly used datasets for the regression tasks. In order to validate our method, and

compare with the existing methods, we select some datasets from UCI Machine Learning Repository, and modify them for the purpose of simulating transfer learning scenarios.

Two datasets “Protein tertiary structure” and “Housing” are considered. The “Protein tertiary structure” contains nine input variables to predict the RMSD-size of the residue, and the dataset is divided into two sub datasets as source and target for the purpose of transfer learning. In the dataset “Housing”, there are six features and the output is the “MEDV”. Because it is difficult to determine the numbers of clusters for the high-dimensional datasets, we adopt a brute-force way to try several different numbers of clusters and choose the one with the best performance. Table 4 gives the results of the above two datasets.

The large values in “ M^s on H_U ” in Table 4 indicate the poor performance of source model in solving the target tasks. And the results in “ M^t on H_U ” validate the effectiveness of our method. We find that no obvious trend is shown with a change in the number of fuzzy rules. So, in the practical situations, we adopt the strategy of going through all numbers in the given range, and select the number of rules with best performance when determining the number of fuzzy rules is difficult.

Table 4. Results for real-world datasets

Protein tertiary structure			Housing		
c	M^s on H_U	M^t on H_U	c	M^s on H_U	M^t on H_U
8	50.88±27.15	6.00±0.01	5	1.40±0.71	0.19±0.00
9	48.90±37.85	5.93±0.01	6	3.11±0.41	0.22±0.01
10	43.32±87.07	6.10±0.01	7	2.41±0.21	0.15±0.00
11	36.84±23.49	5.90±0.01	8	2.51±0.25	0.15±0.01
12	54.41±15.00	5.98±0.00	9	1.60±1.14	0.15±0.00

5 Conclusions and Future Work

This work explores the knowledge contained in the unlabeled target data to improve the performance of the constructed model in solving the domain adaptation problems in regression tasks. The results validate our fuzzy method. Also, the low RMSE in real-world datasets shows the ability of our method in dealing with practical problems.

This method, however, exists a limitation that it works in the situation that the partition of data is obvious. The utilization of unlabeled target data does not show a significant advantage when the boundaries of the clusters in data are ambiguous. How to expand the application scope of our method and explore more information from unlabeled target data will be considered in the future work.

Acknowledgment

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

References

1. Nasrabadi, N. M.: Pattern Recognition and Machine Learning. *Journal of Electronic Imaging*, vol. 16, no. 4, 049901 (2007)
2. Lu, J., Xuan, J., Zhang, G., and Luo, X.: Structural Property-Aware Multilayer Network Embedding for Latent Factor Analysis. *Pattern Recognition*, 76, pp.228-241 (2018)
3. Pan, S. J., and Yang, Q.: A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345-1359 (2010)
4. Lim, C.-H., Wan, Y., Ng, B.-P., and See, C.-M. S.: A Real-Time Indoor WiFi Localization System Utilizing Smart Antennas. *IEEE Transactions on Consumer Electronics*, vol. 53, no. 2 (2007)
5. Xu, J., Ramos, S., Vázquez, D., and López, A. M.: Domain Adaptation of Deformable Part-Based Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 12, pp. 2367-2380 (2014)
6. Long, M., Wang, J., Cao, Y., Sun, J., and Philip, S. Y.: Deep Learning of Transferable Representation for Scalable Domain Adaptation. *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 8, pp. 2027-2040 (2016)
7. Gönen, M., and Margolin, A. A.: Kernelized Bayesian Transfer Learning. In *AAAI*, pp. 1831-1839 (2014)
8. Klenk, M., and Forbus, K.: Analogical Model Formulation for Transfer Learning in AP Physics. *Artificial Intelligence*, vol. 173, no. 18, pp. 1615-1638 (2009)
9. Bengio, Y.: Deep Learning of Representations for Unsupervised and Transfer Learning. In *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, pp. 17-36 (2012)
10. Lu, J., Behbood, V., Hao, P., Zuo, H., Xue, S., and Zhang, G.: Transfer Learning Using Computational Intelligence: A Survey. *Knowledge-Based Systems*, vol. 80, pp. 14-23 (2015)
11. Shao, L., Zhu, F., and Li, X.: Transfer Learning for Visual Categorization: A Survey. *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 5, pp. 1019-1034 (2015)
12. Zuo, H., Zhang, G., Pedrycz, W., Behbood, V., and Lu, J.: Fuzzy Regression Transfer Learning in Takagi–Sugeno Fuzzy Models. *IEEE Transactions on Fuzzy Systems*, vol. 25, no. 6, pp. 1795-1807 (2017)
13. Zuo, H., Zhang, G., Pedrycz, W., Behbood, V., and Lu, J.: Granular Fuzzy Regression Domain Adaptation in Takagi-Sugeno Fuzzy Models. *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 2, pp. 847-858 (2017)
14. Rasmussen, C. E.: The Infinite Gaussian Mixture Model. In *Advances in Neural Information Processing Systems*, pp. 554-560 (2000)