Text Categorization by Fuzzy Domain Adaptation

Vahid Behbood, Jie Lu, Guangquan Zhang
Decision Systems & E-Service Intelligence Research Laboratory
Centre for Quantum Computing and Intelligent System
Faculty of Engineering and Information Technology
University of Technology Sydney
POBOX123, Broadway, NSW 2007, Australia
Vahid.behbood, Jie.lu & Guangquan.zhang@{uts.edu.au}

***Abstract*- Machine learning methods have attracted attention of researches in computational fields such as classification/categorization. However, these learning methods work under the assumption that the training and test data distributions are identical. In some real world applications, the training data (from the source domain) and test data (from the target domain) come from different domains and this may result in different data distributions. Moreover, the values of the features and/or labels of the data sets could be non-numeric and contain vague values. In this study, we propose a fuzzy domain adaptation method, which offers an effective way to deal with both issues. It utilizes the similarity concept to modify the target instances' labels, which were initially classified by a shift-unaware classifier. The proposed method is built on the given data and refines the labels. In this way it performs completely independently of the shift-unaware classifier. As an example of text categorization, 20Newsgroup data set is used in the experiments to validate the proposed method. The results, which are compared with those generated when using different baselines, demonstrate a significant improvement in the accuracy.**

Keywords—Domain Adaptation; Fuzzy Sets; Classification; Text Categorization;

I. INTRODUCTION

Although machine learning has attracted attention of researchers in computational fields such as classification and clustering most learning models such as neural networks and support vector machines work under a common assumption that the training data and test data exhibit the same distribution [1]. When the distribution of test data changes, the learning models need to be rebuilt and retrained from scratch using new training data. For example, in the past years, there are a large number of textual data on the Web such as news reports that were written in formal style. But recently, blogs have been emerging, and their owners begin to write their posts in a style increasingly different from what they read in news reports. Past labeled news data thus cannot be used to reliably classify blog articles, since the usage of vocabulary becomes different in blog articles from news articles. In many real world applications such as text categorization, collecting new training data and retraining the learning model is very expensive or practically not feasible. It would be useful if the data and knowledge gained in different domains could be utilized to assist in the formation of the current learning model.

A new framework of machine learning called Transfer Learning was emerged to handle this issue under a variety of names, such as Learning to Learn, Life-long Learning, Meta Learning, Multi-task Learning and Domain Adaptation [1]. The study of transfer learning has been inspired by human abilities to utilize previously-acquired knowledge to solve new, similar but not identical problems more quickly and efficiently than if this form of knowledge were not available. Transfer learning, which is different from semi-supervised algorithms [2-6], can handle the situation where the domains of training and test data sets are different [7]. However, current transfer learning methods still have some drawback to improve: 1) there is a reliance on statistical models in current transfer learning methods with probabilistic assumptions (e.g., about specific probability distribution functions) that may be difficult to satisfy, and subsequently it could be difficult to achieve highly accurate prediction in some real-world applications; 2) Existing transfer learning methods only consider features and labels whose values are numeric or assume a single value from a discrete set of values of attribute; this assumption could be viewed as a serious impediment in presence of uncertainty; 3) in the existing transfer learning method developments, one attempts to solve the domain adaptation problem by adjusting the decision boundaries and models using global learning; however, this makes the methods highly dependent on the shift-unaware classifier. To address and overcome these limitations, in this study, we propose a Fuzzy Domain Adaptation (FDA) method and investigate its applicability to the problem of text categorization. The FDA method can handle situations of data uncertainty in which the features are vague values and the outputs must provide flexibility and explanatory results to solve the problem appropriately. The key aspect of originality comes with the fact that the domain adaptation problem is solved through refining the fuzzy initial labels in the target domain by similarity-based local learning. The efficiency of the fuzzy set-based approach and the local learning (using fuzzy similarity) for the problem of domain adaptation has been quantified as well.

The main idea behind the proposed FDA method is to explore the most similar instances in a set of mixture domains of the training and test data and treat them as a bridge to transfer the feature distribution from the source domain to the target domain. The label values of these instances are utilized to refine the initial target instances' labels which are reported by a given classifier, referred to as a *shift-unaware*. Using label refinement instead of model adjustment makes the FDA method completely independent from the *shift-unaware* model.

20Newsgroup data set are used for benchmarking the FDA method against the three machine learning models: Support Vector Machine [8]; Multi Layer Perceptron Neural Network [9]; and Fuzzy Neural Network [10] along with the existing domain adaptation method [8]. The results demonstrate the superior performance of the proposed algorithm and show the significant role of a fuzzy set-based approach and local learning in accuracy enhancement.

The rest of paper is organized as follow. In Section II, some preliminaries concepts including the definition of domain adaptation and related works are given. Section III proposes the FDA method and Section IV describes the experimental illustration and results. Section V concludes this paper and discusses future researches.

II. PRELIMINARIES AND RELATED WORKS

The definition and related notation of terms and concepts that used throughout paper are introduced in this section, following which the categories of transfer learning and related works are described.

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

Definition 2 (Task) [1]: A task is denoted by $T = \{Y, f(\cdot)\}$ where $Y = \{y_1, \dots, y_m\}$ is a label space and $f(\cdot)$ is an objective predictive function which is not observed and has to be learned by pairs (x_i, y_i). The function $f(\cdot)$ can be used to predict the corresponding label, $f(x_i)$, of a new instance x_i. From a probabilistic viewpoint, $f(x_i)$ can be expressed as $P(y_i|x_i)$ where P is the probability of label y_i for given instance x_i.

Definition 3 (Transfer Learning) [1]: Given a source domain D_s and learning task T_s a target domain D_t, and learning task T_t,, transfer learning aims to improve the 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 $T_s \neq T_t$. In addition, there are some explicit or implicit relationships among the feature spaces of two domains, such that we imply that the source domain and target domain are related. It should be mentioned that when the target and source domains are the same ($D_s = D_t$) and their learning tasks are also the same ($T_s = T_t$), the learning problem becomes a traditional machine learning problem.

Definition 4 (Multi task learning) [1]: In the above definition, the condition $T_s \neq T_t$ implies that either $Y_s \neq Y_t$ or $f_s(\cdot) \neq f_t(\cdot)$ or both. This condition is called multi task learning.

Definition 5 (Transductive transfer learning) [1]: Similarly, the condition $D_s \neq D_t$ implies that either $F_s \neq F_t$ or $P_s(X) \neq P_t(X)$. This condition called transductive transfer learning.

Definition 6 (Domain Adaptation) [1]: Domain Adaptation is a category of transductive transfer learning in which $F_t = F_s$ but $P_t(X) \neq P_s(X)$.

The studies that aim to solve the domain adaptation problem can be categorized into two groups [1]: (i)Transferring the knowledge of instances: this approach is motivated by the importance of samples and an attempt is made to find an optimum weight for each instance to learn a precise model for the target domain. Papers in this category can be found in a recently published book by Quionero-Candela et al. [11]; (ii)Transferring the knowledge of feature representation: this approach focuses on the feature space and attempts to extract and/or convert relevant features which reduce the difference between the domains. Blitzer et al. [12-14] proposed an SCL algorithm to define pivot features on the target domain from both domains and then used unlabeled instances from the target to create the classification model. Dai et al. [15] proposed a co-clustering based algorithm to propagate the label information across domains. Xue et al. [16] presented a cross-domain text classification known as TPLSA to integrate target instances and source instances into a unified probabilistic model. This study is in line with this category and focuses on the feature representation and distribution for transfer knowledge between domains.

III. FUZZY DOMAIN ADAPTATION METHOD

Given D_s is the source domain with fuzzy feature sets $F^s = \{f_1^s, \dots, f_m^s\}$ and D_t is the target domain with fuzzy features $F^t = \{f_1^t, \dots, f_m^t\}$, where f_i is a fuzzy trapezoidal-shaped membership functions for each feature. It is assumed that the number of these features is the same for both D_s and D_t the target and source domains but the membership function of these fuzzy sets may be different. This assumption implies a transductive transfer learning problem in which the feature space is the same but the distribution of data is different (domain adaptation). Figures 1 and 2 show example of fuzzy feature (f_i) in source and target domains respectively. There are five membership functions (linguistic terms) distributed over the range of (-2, 4) in both domains, but the membership functions are different.

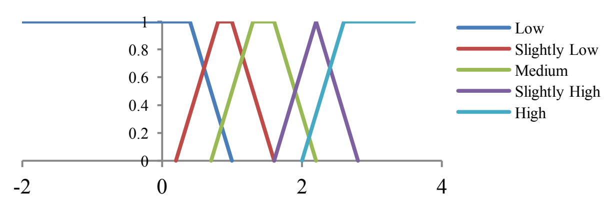


Fig. 1. Fuzzy Feature f_i in Source Domain

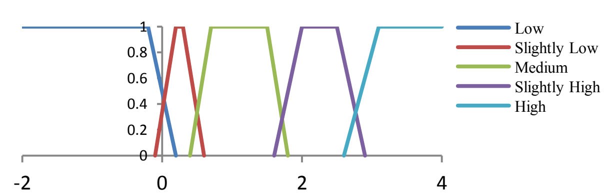


Fig. 2. Fuzzy Feature f_i in Target Domain

$Y = \{Y_1, \dots, Y_L\}$ is the predictive fuzzy label set which is the same for both domains D_s and D_t . For instance, we have $L = 2, Y_1 = (Positive)$ and $Y_2 = (Negative)$. $g(\cdot)$ is a learning-based classifier such as a neural network. If the learning-based classifier $g(\cdot)$ has been trained by the labeled data coming from the source domain, we will call it a shift-unaware classifier and denote it by $g(\cdot)|D_s$. Thus, $g(x)|D_s = \left(\mu_{Y_1}(x), \dots, \mu_{Y_L}(x)\right)$ will be the vector of membership values of x in each class computed by the shift-unaware classifier. Formally, we give the following definition:

Let matrix G denote the unrefined label matrix where g_{ij} is the membership value of a given instance x_i in class j which is computed by shift-unaware classifier $g(\cdot)|D_s$. S is the fuzzy similarity function. Let M denote the similarity matrix where m_{ij} measures the similarity between the given instances x_i and x_j using $S(x_i, x_j)$. Let MR^1 denotes the refined label matrix in first iteration where mr_{ij}^1 is the refined membership value of given instance x_i in class j after first iteration of the refinement. The following expression describes the first iteration of refinement:

$$mr_{ij}^1 = \alpha \left(\frac{\sum_{x_0 \in N_i^D} S(x_i,x_0)\left(\mu_{Y_j}(x_i) - \mu_{Y_j}(x_0)\right)}{|N_i^D|} \right) + (1-\alpha) g_{ij} \qquad (1)$$

where N_i^D is a set of most similar instances to instance x_i that can be extracted from a given domain D using the similarity matrix M. Given α is the parameter, which is used to specify the impact of refinement. According to the refined Equation 1, we calculate the difference between the label values of most similar instances and that of the given instance $\left(\mu_{Y_j}(x_i) - \mu_{Y_j}(x_0)\right)$. This is multiplied by the similarity value between most similar instances and the given instance $S(x_i, x_0)$ to amplify the influence of more similar instances on refinement. Finally, the average value is used to refine the unrefined label values g_{ij} by an impact factor of α. To compute the refined value in the next iterations, the label values computed in the prior iteration are applied as follows:

$$mr_{ij}^{(t+1)} = \alpha \left(\frac{\sum_{x_0 \in N_i^D} S(x_i,x_0)\left(mr_{ij}^{(t)} - mr_{x_0 j}^{(t)}\right)}{|N_i^D|} \right) + (1-\alpha) g_{ij} \qquad (2)$$

According to (2), the refined labels of instances in the previous iterations are used to further adjust the label values in the current iteration. Using this approach leads to a mutual reinforcement relationship between instances in domains that can help to transfer the label pattern from the source domain to the target domain and consequently augment the accuracy.

As mentioned previously, D_s is with fuzzy feature sets $F^s = \{f_1^s, \dots, f_m^s\}$ and D_t is with fuzzy features $F^t = \{f_1^t, \dots, f_m^t\}$, where f_i is a fuzzy trapezoidal membership function for each feature. The steps of the FDA method being organized into two phases: Phase 1 is a preprocessing phase completed to represent (encode) numeric input in terms of the fuzzy sets (reference fuzzy sets) defined in the given input variable, compute the initial label values using a shift-unaware classifier $g(\cdot)$ and calculate the similarity matrix; Phase 2 is the refinement phase in which we apply the proposed refinement Equation (2) in Step 2-3.

Phase 1: Preprocessing
Input:

- Source domain: D_s
- Target domain: D_t
- Fuzzy label space: Y
- Shift-unaware classifier: $g(\cdot)$
- Fuzzy similarity function: $S(\cdot)$

Output:

- Unrefined label matrix for instances in target domain: G

Step 1-1: Use singleton fuzzifier to encode numeric input of instances from both domains.

$$\mu_{\tilde{x}_i}(\tilde{x}_i) = \begin{cases} 1, & \text{if } \tilde{x}_i = x_i \\ 0, & \text{Otherwise} \end{cases}$$

Step 1-2: Perform antecedent matching of fuzzyfied inputs $x_i \in D_s$ and D_t against fuzzy features F^s and F^t respectively, respectively. The input membership value of each fuzzy set is computed as follows:

$$\mu_{\tilde{f}_k}(\tilde{x}_i) = \begin{cases} 0, & \text{if} \quad x_i \le l_{\tilde{f}_k} \\ \frac{x_i - l_{\tilde{f}_k}}{u_{\tilde{f}_k} - l_{\tilde{f}_k}}, & \text{if} \quad l_{\tilde{f}_k} \le x_i \le u_{\tilde{f}_k} \\ 1, & \text{if} \quad u_{\tilde{f}_k} \le x_i \le v_{\tilde{f}_k} \\ \frac{r_{\tilde{f}_k} - x_i}{r_{\tilde{f}_k} - u_{\tilde{f}_k}} & \text{if} \quad v_{\tilde{f}_k} \le x_i \le r_{\tilde{f}_k} \\ 0, & \text{if} \quad x_i \ge r_{\tilde{f}_k} \end{cases}$$

Step 1-3: Compute the similarity matrix using the fuzzy similarity function $S(\cdot)$:

For i = 1 to $|D_s| + |D_t|$
  For j = 1 to $|D_s| + |D_t|$
    $m_{ij} = S(x_i, x_j)$
  Next j
Next i

Step 1-4: Train shift-unaware classifier $g(\cdot)|D_s$ by labeled data of source domain.

Step 1-5: Calculate the unrefined label matrix for target domain instances (G) using $g(\cdot)|D_s$ as follows:

For i = 1 to $|D_t|$
  For j = 1 to L
    $g_{ij} = [g(x_i)|D_s]_j = \left(\mu_{Y_j}(x_i)\right)$
  Next j
Next i

Phase 2: Refinement
Input:

- Source domain: D_s
- Target domain: D_t
- Fuzzy label space: Y
- Fuzzy similarity function: $S(\cdot)$
- Unrefined Label Matrix: G
- Impact tradeoff parameter: α
- Number of most similar instances: K
- Number of steps of refinement: n

Output:

- Refined Label matrix for instances in target domain MR^n.

For $w = 1$ to n
Step 2-1: Create the mixture domain D_w combination of source and target domain as follows:

$$D_w = (1 - {}^{w}/_{n})|D_s| + {}^{w}/_{n}\,|D_t|$$

Step 2-2: Find K most similar instances ($N_i^{D_w}$) for each target instance. These instances are extracted from mixture domain D_w.

For $i = 1$ to $|D_t|$
  $N_i^{D_w} = \{n_i = \mathrm{argmax}_j m_{ij},\ n_i \in D_w\}$
Next i

Step 2-3: Refine the fuzzy label for each target instance.

Repeat t
For $i = 1$ to $|D_t|$
  For $j = 1$ to L

$$mr_{ij}^{(w)}(t) = \alpha\left(\frac{\sum_{x_0 \in N_i^{D_w}} S(x_i, x_0)\left(mr_{ij}^{(w-1)}(t-1) - mr_{x_0 j}^{(w-1)}(t-1)\right)}{|N_i^{D_w}|}\right) + (1-\alpha) g_{ij}$$

  Next j
Next i
Until MR^w converges
Next w

The refinement expression (2) is applied in the proposed FDA method in Step 2-3. The refinement is based on the fact that the label of the most similar instances to the target instance is used to modify the initial label of the target instance, which was initialized by the shift-unaware model. As a result, the refined fuzzy label matrix for all unlabeled instances of target domain MR^n is formed as follows:

$$MR^n = \begin{bmatrix} mr_{11}^n & \cdots & mr_{1L}^n \\ \vdots & \ddots & \vdots \\ mr_{|D_t|1}^n & \cdots & mr_{|D_t|L}^n \end{bmatrix} \tag{3}$$

Each row of this matrix shows the membership value of one instance to all label classes. To find the final label for each instance, the expression is used:

$$\text{Label}\,(x_i) = \mathrm{argmax}_i\{mr_{ij}^n | j = 1, \dots, L\}.$$

IV. EXPERIMENTS

In this section, we report on the experiments that have been performed using the widely used 20Newsgroup data set (http://people.csail.mit.edu/jrennie/20Newsgroups/). Different settings of the proposed algorithm are described and the data set specifications are explained. The baselines are also introduced for benchmarking. Finally, empirical results are analyzed.

A. Setting

The algorithm was realized by using four different settings based on different mixture domains D_w and the number of steps in Phase 2 (Refinement) of the algorithm. These settings are presented in Table I. All settings are divided into four categories. Categories 1 and 2 refer to the settings with one and two steps of refinement, respectively. Category 3 and 4 contains the settings with three and $\boldsymbol{n}$ steps of refinement. Hence, each category indicates the number of iterations of the refinement process with different possible mixture domains. Table I shows that Category 4 has two similar setup with $\boldsymbol{n}$ steps of refinement. However, in one setup 4-2, we use a smaller number of labeled instances of the target domain in mixture domains to examine the influence of labeled target data on the performance of the proposed method.

B. Data set and prepration

We validate the proposed method by using a commonly used data set, namely 20Newsgroup. The different settings mentioned above are investigated. This data collection was not originally designed for transfer learning, so some modification was necessary to make the distribution between the training data and the test data different. The data set has a two-level hierarchical structure. Suppose A and B are two root categories in the data collection, and A1, A2 and B1, B2 are sub-level categories of A and B, respectively. We form the training and test data in the way. Let A.A1, B.B1 be the positive and negative examples in the training data respectively. Let A.A2, B.B2 be the positive and negative examples in the test data, respectively. Thus, the target categories are fixed, being A and B, but the distributions of the training data and the test data are different yet still similar enough for the evaluation of the proposed algorithm in transfer learning. There are seven top level categories, while three of them have no sub-categories. We compose six data sets from the remaining four categories. The detailed composition of these data sets is provided in Table II.

We make some preprocessing on the raw data by including turning all letters into lowercases, stemming words by the Porter stemmer [17], and removing all stop words. According to [18], the DF Thresholding can achieve comparable performance to Information Gain or CHI, but it is much easier to implement and less costly both in time and space requirements. Hence we use it to cut down the number of words/features and speed up the classification. The words that occur in fewer than three documents are removed. Each document is then converted into a bag of-words presentation in the remaining feature space. Each value of the feature is the term frequency of that word in the document, weighted by its IDF (log N/DF). To examine the performance of the FDA method, we select three different shift-unaware classifiers: Fuzzy Neural Network (FNN) [8]; Support Vector Machine (SVM) [9]; Multi Layer Perception Neural Network (MLPNN) [10]. Discrete Incremental Clustering (DIC) [19], which is a novel self-organizing clustering technique, is applied to create the fuzzy features. We use the fuzzy similarity/dissimilarity measure presented in [20-21] in the proposed FDA method.

TABLE I. DIFFERENT SETTINGS OF THE FDA METHOD

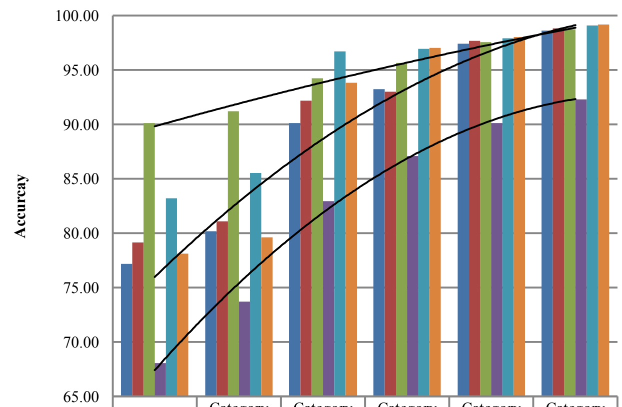
Category	N.O Steps (Iterations)	Mixture Domain (D_w)
1	1	D_t
2	2	$D_s \cup D_t$
3	3	$D_s \cup D_t$
4-1	n	$D_s \cup D_t$
4-2	n	$D_s \cup D_t$ *

TABLE II. 20NEWSGROUP DATA COLLECTION AND ITS DETAILED COMPOSITION DATA SETS

Data set	Train/Test data	Positive	Negative	Number of samples
1	Train	rec.autos rec.motorcycles	talk.politics.guns talk.politics.misc	3669
	Test	rec.sport.baseball rec.sport.hockey	talk.politics.mideast talk.religion.misc	3561
2	Train	rec.autos rec.sport.baseball	sci.med sci.space	3961
	Test	rec.motorcycles rec.sport.hockey	sci.crypt sci.electronics	3954
3	Train	comp.graphics comp.sys.mac.hardware comp.windows.x	talk.politics.mideast talk.religion.misc	4482
	Test	comp.os.ms-windows.misc comp.sys.ibm.pc.hardware	talk.politics.guns talk.politics.misc	3652
4	Train	comp.graphics comp.os.ms-windows.misc	sci.crypt sci.electronics	3930
	Test	comp.sys.ibm.pc.hardware comp.sys.mac.hardware comp.windows.x	sci.med sci.space	4900
5	Train	comp.graphics comp.sys.ibm.pc.hardware comp.sys.mac.hardware	rec.motorcycles rec.sport.hockey	4904
	Test	comp.os.ms-windows.misc comp.windows.x	rec.autos rec.sport.baseball	3949
6	Train	sci.electronics sci.med	talk.politics.misc talk.religion.misc	3374
	Test	sci.crypt sci.space	talk.politics.guns talk.politics.mideast	3828

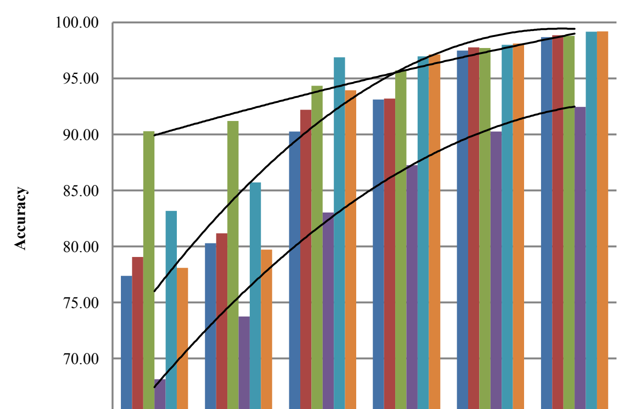
C. Empirical results analysis

The experimental results show that in all cases the proposed algorithm improves accuracy. As shown in Figures 3, 4 and 5, the greatest increase of accuracy is noted for Categories 4-1 and 4-2 with multiple iterations of refinement and mixture domains of target and source domains in each step. This demonstrates that multi-step refinement can significantly improve accuracy and produces better results compared to other settings with fewer refinement iterations. Thus, it can be concluded that the number of refinement steps has influence on performance and becomes beneficial in boosting accuracy. In what follows, we focus on the Category 4-2 of the proposed algorithm, which is the most successful one among the alternatives being considered. Its results are compared with the unrefined results. Figure 3 shows the accuracy of different settings of the FDA method using SVM on all 20Newsgroup data sets compared with the accuracy of the unrefined results. In all data sets, the proposed method improves the accuracy, particularly in data sets 5 and 6, in which the relative increases are 26.9% and 27.1% respectively. The average relative increase of accuracy in the Category 4-2 over all the six data sets is 25%. Figure 4 reports the results of the proposed method with MLPNN viewed as a shift-unaware classifier. The highest relative enhancements of the accuracy are achieved on data sets 5 and 6, being 26.5% and 26.8% respectively. The average relative increase in the category 4-2 is 24.8%. Figure 5 demonstrates the result of the refinement of the FNN results using the proposed method. The greatest relative increases in accuracy are achieved for data sets 5 and 6, with 26.9% and 27.3% respectively. The average relative increase in Category 4-2 is 25.6%.



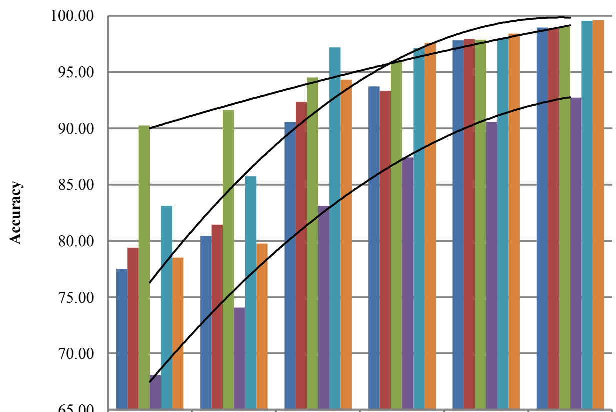
	SVM	Category 1	Category 2	Category 3	Category 4-1	Category 4-2
Data Set 1	77.17	80.17	90.12	93.22	97.40	98.60
Data Set 2	79.16	81.07	92.17	93.00	97.66	98.80
Data Set 3	90.12	91.21	94.23	95.63	97.54	98.73
Data Set 4	68.08	73.71	82.93	87.08	90.12	92.28
Data Set 5	83.20	85.52	96.71	96.92	97.91	99.08
Data Set 6	78.12	79.63	93.81	97.01	98.02	99.18

Fig. 3. Accuracy of the FDA method when using SVM as shift-unaware calssifier under 4 categories



	MLPNN	Category 1	Category 2	Category 3	Category 4-1	Category 4-2
Data Set 1	77.38	80.30	90.27	93.10	97.48	98.68
Data Set 2	79.07	81.18	92.19	93.19	97.76	98.84
Data Set 3	90.29	91.19	94.34	95.70	97.70	98.80
Data Set 4	68.14	73.75	83.04	87.26	90.26	92.45
Data Set 5	83.18	85.71	96.89	96.97	97.98	99.16
Data Set 6	78.09	79.70	93.95	97.14	98.09	99.19

Fig. 4. Accuracy of the FDA method when using MLPNN as shift-unaware calssifier under 4 categories



	FNN	Category 1	Category 2	Category 3	Category 4-1	Category 4-2
Data Set 1	77.49	80.45	90.56	93.72	97.80	98.95
Data Set 2	79.38	81.44	92.34	93.32	97.91	98.83
Data Set 3	90.25	91.60	94.52	95.89	97.87	99.02
Data Set 4	68.09	74.07	83.12	87.41	90.56	92.72
Data Set 5	83.12	85.74	97.17	97.12	97.99	99.53
Data Set 6	78.50	79.75	94.30	97.57	98.40	99.58

Fig. 5. Accuracy of the MFBRDA algorithm when using FNN as shift-unaware classification model under 9 settings

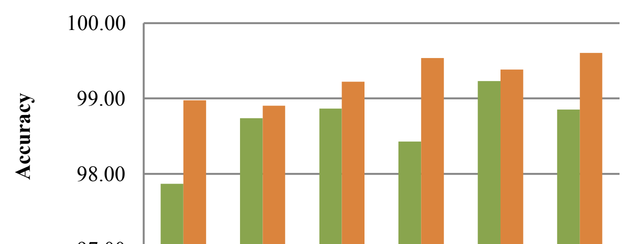
Additionally, we compare the performance of the proposed FDA method (Category 4-2) with another Domain Adaptation method (DA) [8]. This method, which considers the features with numeric values and uses the *Cos* function expressed as ($dist(i,j) = 1 - cos(di,dj)$) as the corresponding distance. The results of comparison demonstrate an impact of the fuzzy set-based approach to the quality of the obtained results. In the benchmark, three different shift-unaware classifiers; SVM, MLPNN and FNN are used to determine the initial labels. Figures 6, 7 and 8 show the benchmark results by reporting the accuracy of FDA and DA methods on the 20Newsgroup data sets. The results clearly show that the FDA method outperforms the DA method in all data sets when using different classifiers. For instance, the average increases in accuracy achieved by FDA on data sets 3, 4 and 6 are 1.4%, 1.44% and 1.41%, respectively. Similarly, the average increases of accuracy gained by FDA using SVM, MLPNN and FNN classifiers are equal to 1.2%, 1.1% and 1.0% , respectively. All in all, the fuzzy set-based approach applied to the FDA algorithm significantly improves the refinement performance and boosts accuracy.

V. CONCLUSION AND FUTURE STUDY

The research challenge in this study was to develop a domain adaptation algorithm, which can be made independent of the shift-unaware classifier and work with any given model. Also, the objective of this study was to develop a domain adaptation algorithm that would be able to handle the uncertainty of data and deal with vague (non-numeric) values of the features and class labels. To cope with the two issues, the FDA method was proposed using a fuzzy similarity-based local learning approach. The method focuses on the modification of the instances' labels in the target domain using nine different settings and engaging fuzzy set representation of the features to improve accuracy of classification. Three shift-unaware classifier were applied to determine the initial labels for unlabeled target instances. The obtained experimental results show that the proposed FDA method brings about a remarkable improvement in performance. A significant increase in accuracy has been reported, in particular when the method used more iterations and utilizes a few labeled target data along with source data and unlabeled target data.

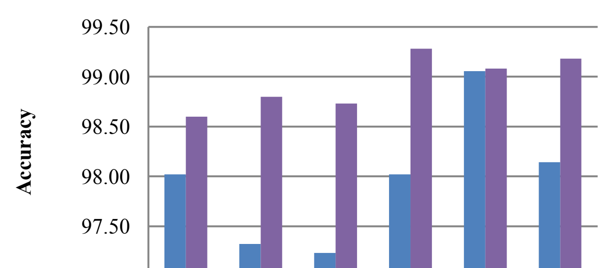
It is worth noting that, in comparison with an existing refinement method called DA, the FDA applies fuzzy sets to modify the initial prediction and according to the empirical results, it substantially outperforms the DA method. The FDA is independent from the prediction model and can be applied with other methods. We showed that the FDA can successfully transfer knowledge where we are faced with insufficient number of recent training data

Our future studies will focus on three tasks. One is to use other prediction or classification models such as fuzzy case base reasoning and fuzzy rule-based learning models to realize transfer learning. Another one is to develop a method, based on the proposed method, which can extract the relevant features to reduce the difference between domains. Finally, an interesting and promising direction could be to examine the performance of the proposed method in contrast to other transfer learning methods using different real-world data sets applications.



	Data Set 1	Data Set 2	Data Set 3	Data Set 4	Data Set 5	Data Set 6
FNN-DA	97.87	98.74	98.86	98.43	99.23	98.85
FNN-FDA	98.97	98.90	99.22	99.54	99.38	99.60

Fig. 6. Accuracy of FDA and DA methods using FNN with 6 data sets



	Data Set 1	Data Set 2	Data Set 3	Data Set 4	Data Set 5	Data Set 6
SVM-DA	98.02	97.32	97.23	98.02	99.05	98.14
SVM-FDA	98.60	98.80	98.73	99.28	99.08	99.18

Fig. 7. Accuracy of FDA and DA methods using SVM with 6 data sets

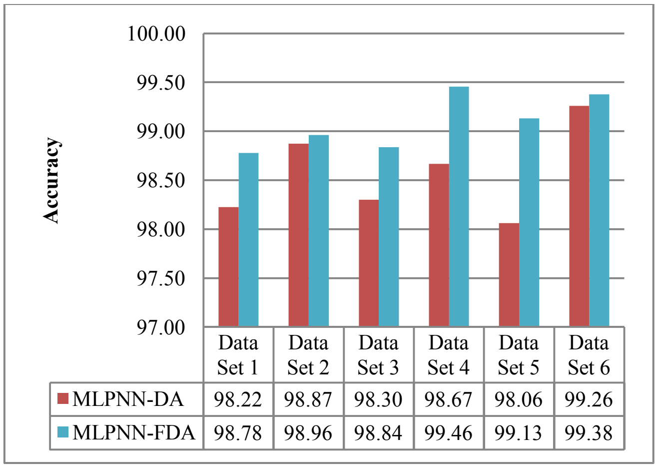


Fig. 8. Accuracy of FDA and DA methods using MLPNN with 6 data sets

ACKNOWLEDGMENT

The work presented in this paper was supported by Australian Research Council (ARC) under the Discovery Projects DP088739 and DP110103733.

REFERENCES

[1] S. J. Pan and Q. Yang, "A survey on transfer learning," IEEE Transactions on Knowledge and Data Engineering, vol. 22, pp. 1345-1359, 2010.

[2] X. Zhu, "Semi-Supervised Learning Literature Survey," Computer Science, University of Wisconsin-Madiso,, Madison, WI2005.

[3] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell, "Text classification from labeled and unlabeled documents using EM," Machine Learning, vol. 39, pp. 103-134, 2000.

[4] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in Eleventh Annual Conference on Computational Learning Theory, Madison, WI, 1998, pp. 92-100.

[5] T. Joachims, "Transductive inference for text classification using support vector machines," in Sixteenth International Conference on Machine Learning, Bled, Slovenia, 1999, pp. 200-209.

[6] V. Behbood, J. Lu, and G. Zhang, "Long term bank failure prediction using fuzzy refinement-based transductive transfer learning," in IEEE International Conference on Fuzzy Systems, Taipei, Taiwan, 2011, pp. 2676-2683.

[7] G. P. C. Fung, J. X. Yu, L. Hongjun, and P. S. Yu, "Text classification without negative examples revisit," IEEE Transactions on Knowledge and Data Engineering, vol. 18, pp. 6-20, 2006.

[8] D. Xing, W. Dai, G.-R. Xue, and Y. Yu, "Bridged refinement for transfer learning," in 11th European Conference on Principles and Practice of Knowledge Discovery in Databases, Warsaw, Poland, 2007, pp. 324-335.

[9] C.-T. Lin and C. S. G. Lee, Neural Fuzzy Systems: A Neuro-Fuzzy Synergism to Intelligent Systems. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[10] V. Behbood, J. Lu, and G. Zhang, "Adaptive Inference-based learning and rule generation algorithms in fuzzy neural network for failure prediction," in IEEE International Conference on Intelligent Systems and Knowledge Engineering (ISKE), Hangzhou, China, 2010, pp. 33-38.

[11] J. Quionero-Candela, M. Sugiyama, A. Schwaighofer, and N. Lawrence, Dataset Shift in Machine Learning. Cambridge: The MIT Press, 2009.

[12] J. Blitzer, R. McOnald, and F. Pereira, "Domain adaptation with structural correspondence learning," in Conference on Empirical Methods in Natural Language Processing, Sydney, Australia, 2006, pp. 120-128.

[13] J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. Wortman, "Learning bounds for domain adaptation," in Twenty-First Annual Conference on Neural Information Processing Systems, Cambridge, MA, 2007, pp. 245-252.

[14] J. Blitzer, M. Dredze, and F. Pereira, "Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification," in 45th Annual Meeting of the Association of Computational Linguistics, Prague, Czech Republi, 2007, pp. 440-447.

[15] W. Dai, G.-R. Xue, Q. Yang, and Y. Yu, "Co-clustering based classification for out-of-domain documents," in 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,, San Jose, CA, 2007, pp. 210-219.

[16] G.-R. Xue, W. Dai, Q. Yang, and Y. Yu, "Topic-bridged PLSA for cross-domain text classification," in 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Singapore, Singapore, 2008, pp. 627-634.

[17] M. F. Porter, "An algorithm for suffix stripping," Program, vol. 14, pp. 130-137, 1980.

[18] Y. Yang and J. O. Pedersen, "A comparative study on feature selection in text categorization," in Fourteenth International Conference on Machine Learning, 1997, pp. 412-420.

[19] W. L. Tung, C. Quek, and P. Cheng, "GenSo-EWS: A novel neural-fuzzy based early warning system for predicting bank failures," Neural Networks, vol. 17, pp. 567-587, 2004.

[20] W. Wen-June, "New similarity measures on fuzzy sets and on elements," Fuzzy Sets and Systems, vol. 85, pp. 305-309, 1997.

[21] W. Chao, L. Jie, and Z. Guangquan, "Integration of Ontology Data through Learning Instance Matching," in *IEEE/WIC/ACM International Conference on Web Intelligence*, 2006, pp. 536-539.