Effectively Constructing Reliable Data for Cross-Domain Text Classification

Fuzhen Zhuang, Qing He, Zhongzhi Shi

Effectively Constructing Reliable Data for Cross-domain Text Classification

Fuzhen Zhuang, Qing He, and Zhongzhi Shi

The Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China.
{zhuangfz, heq, shizz}@ics.ict.ac.cn

Abstract. Traditional classification algorithms often fail when the *independent* and *identical distributed* (i.i.d.) assumption does not hold, and the cross-domain learning emerges recently is to deal with this problem. Actually, we observe that though the trained model from training data may not perform well over all test data, it can give much better prediction results on a subset of the test data with high prediction confidence. Also this subset of data from test data set may have more similar distribution with the test data. In this study, we propose to construct the reliable data set with high prediction confidence, and use this reliable data as training data. Furthermore, we develop an EM algorithm to refine the model trained from the reliable data. The extensive experiments on text classification verify the effectiveness and efficiency of our methods. It is worth to mention that the model trained from the reliable data achieves a significant performance improvement compared with the one trained from the original training data, and our methods outperform all the baseline algorithms.

Key words: Cross-domain Learning; Reliable Data; EM Algorithm.

1 Introduction

Classification techniques play an important role in intelligent information process, such as the analysis of World-Wide-Web pages, images and so on. This requires the trained models from the training data (also referred as *source domain*) to give correct prediction or classification on unlabeled test data (also referred as *target domain*). Traditional classification techniques, e.g., Support Vector Machine (SVM) [2, 3] and Naïve Bayesian (NBC) [15, 19] and Logistical Regression (LR) [9], are proved to perform very well when the training and test data are drawn from the *independent* and *identical distribution* (i.i.d.). However, this assumption always actually does not hold in real-world applications, since the test data usually come from the information sources with different distribution caused by the sample selection bias, concept drift and so on.

In recent years, cross-domain learning[1] [23] has attracted a great attention, which focuses on the model adaptation from source domain to target domain with distribution mismatch. These works include feature selection learning [4, 29, 21, 25], transfer learning via dimensionality reduction [20, 7], model combination [6, 30] and sample selection bias [26, 28] et al.

[1] Also referred as transfer learning or domain adaptation in the previous research.

Table 1. The Prediction Probabilities for Two-class Classification on 20 Documents (The label in bracket is the true label of document)

		$d_1(c_1)$	$d_2(c_1)$	$d_3(c_1)$	$d_4(c_1)$	$d_5(c_2)$	$d_6(c_2)$	$d_7(c_2)$	$d_8(c_2)$	$d_9(c_2)$	$d_{10}(c_2)$
Classes	c_1	1.00	0.99	0.98	0.985	0.97	0.015	0.01	0.02	0.005	0.025
	c_2	0.00	0.01	0.02	0.015	0.03	0.985	0.99	0.98	0.995	0.975
		$d_{11}(c_2)$	$d_{12}(c_2)$	$d_{13}(c_2)$	$d_{14}(c_2)$	$d_{15}(c_2)$	$d_{16}(c_1)$	$d_{17}(c_1)$	$d_{18}(c_1)$	$d_{19}(c_1)$	$d_{20}(c_1)$
Classes	c_1	0.91	0.12	0.10	0.85	0.89	0.92	0.09	0.20	0.86	0.13
	c_2	0.09	0.88	0.90	0.15	0.11	0.08	0.91	0.80	0.14	0.87

Unlike previous research, our work is motivated by the observation that although the trained model from source domain may give poor performance over all the target domain data, it might perform well on an elaborately selected subset. Let's look at an intuitive example in Table 1, there are prediction probabilities (shown in columns 2, 3, 5 and 6) for classes c_1 and c_2 of 20 documents ($d_1 \sim d_{20}$) given by the model. The documents $d_1 \sim d_{10}$ are with probabilities higher than 0.96 when their labels are predicted as c_1 or c_2, while the rest documents $d_{11} \sim d_{20}$ are with much lower probabilities (lower than 0.93). If we compute the prediction accuracy over all the test documents, we can only obtain a low accuracy 65%. However, if the samples with higher prediction probabilities are selected (e.g., higher than 0.96), we can get a much better performance 90% on this sub data set (e.g., $d_1 \sim d_{10}$). Indeed, this coincides with the human sentiment that people always give higher confidence when they make sure of something, and vice versa they give lower confidence when not sure. Thus, we can trust the prediction results when the test samples with high prediction probabilities. Under these observations, in this paper we propose a two-step method for cross-domain text classification. First, we construct the reliable data set with high prediction confidence from target domain, and then use them as "labeled" data[2] to train a model. Second, we further propose an EM algorithm to refine the model trained from the reliable data set. This is an intuitively appealing fact, the model trained from reliable data set may perform much better than the one from source domain, since the reliable data are selected from target domain and with more similar distribution. The experimental results in Section 4 validate that.

A word for the outline of this paper. Section 2 survey some related works, and followed by the detailed description of the proposed method in Section 3. In Section 4, we evaluate our method on a large amount of experiments on text classification. Finally, we conclude the paper in Section 5.

2 Related Works

Here we summarize some previous works which are mostly related to this paper, including cross-domain learning and learning positive and unlabeled samples.

2.1 Cross-domain Learning

Cross-domain Learning studies how to deal with the classification problem when the source and target domain data obey different distributions. There are many papers appear in recent years, and they can be grouped into three types of techniques used

[2] The reliable data are not really labeled data, since their labels are predicted by the trained model from source domain.

for knowledge transfer, namely feature selection based [11, 4, 29], feature space mapping [20, 22, 7], weight based [5, 6, 10].

For the feature selection based methods, Jiang et al. [11] developed a two-step feature selection framework for domain adaptation. They first selected the general features to build a general classifier, and then considered the unlabeled target domain to select specific features for training target classifier. Dai et al. [4] proposed a Co-clustering based approach for this problem. In this method, they identified the word clusters among the source and target domains, via which the class information and knowledge propagated from source domain to target domain. Feature space mapping based methods are to map the original high-dimensional features into a low-dimensional feature space, under which the source and target domains comply with the same data distribution. Pan et al. [20] proposed a dimensionality reduction approach to find out this latent feature space, in which supervised learning algorithms can be applied to train classification models. Gu et al. [7] learnt the shared subspace among multiple domains for clustering and transductive transfer classification. In their problem formulation, all the domains have the same cluster centroid in the shared subspace. The label information can also be injected for classification tasks in this method. For the weight based methods, Jiang et al. [10] proposed a general instance weighting framework, which has been validated to work well on NLP tasks. Gao et al. [6] proposed a dynamic model weighting method for each test example according to the similarity between the model and the local structure of the test example in the target domain.

The most related work is [28]. Instead of selecting reliable data set from target domain, they used the labeled data from target domain to select useful data points in source domain, and then combined them with a few labeled data from target domain to build a good classifier. The main difference from our work is that, they needed some labeled data from target domain, while the target domain data in our problem are totally unlabeled.

2.2 Learning from Positive and Unlabeled Samples

The research of learning from positive and unlabeled (*LPU*) samples focuses on the application scenarios that there are only labeled positive samples but not any negative ones. Several techniques were proposed in [18, 14, 17, 16, 27, 8].

Most of these methods take a two-step strategy. First, they adopted the techniques, e.g., Rocchio algorithm [24], Support Vector Machine (SVM) [3] and Naïve Baysian method [19] et al., to extract some reliable negative examples from the unlabeled data set, and then used the positive and likely negative samples to train a model. Second, the EM algorithm or iterative SVM were used to update the model. For example, Liu et al. [18] proposed a S-EM method for *LPU* learning, in which they first selected some documents from positive samples as "spy" documents to select more reliable negative data, and then used EM algorithm to build the final classifier. Although we also develop a two-step method and is similar with the techniques in *LPU* learning, there are two main differences from *LPU* learning. 1) Our method is to find reliable data set from target domain data for cross-domain learning, in which there are not any labeled data from target domain and the distributions of labeled source domain and unlabeled target domain are different, also *LPU* learning aims to build a binary classifier; 2) In the second

step of our approach, the prediction probabilities of all target domain data (including the reliable data set) are updated during the EM iteration, while in *LPU* learning the probabilities of labeled positive samples retain unchanged. We adapt S-EM method to select the reliable data for our problem setting, but the performance is poor. To the best of our knowledge this is the first time to construct reliable data set for cross-domain learning, and the experimental results verify its effectiveness.

Another related work is Co-training [1]. Co-training assumed there were two views of feature set for data instances, and either view of the examples would be sufficient for learning if given enough labeled data. Also in Co-training algorithm, the unlabeled samples were selected as labeled data according to the consensus prediction of the two classifiers learnt from two views of labeled data. Instead, in our method we select the unlabeled data with high confident prediction as reliable data set.

3 The Proposed Algorithms

In this paper we propose two-step approaches for cross-domain classification: the first step is to select the reliable data set from target domain data which are predicted with high confidence by the trained model from source domain; then secondly, an EM algorithm is used to refine the model trained on the reliable data.

It is worth to note that selecting reliable data set is very important for our method, so we adopt the state-of-the-art supervised classifiers Logistical Regression [9] and Naïve Baysian method [19], which not only can produce probabilistic predictions, but also are approved to perform very well when the distributions between training and test data are the same.

Given the source and target domains, denoted as $\mathcal{D} = \{\mathcal{D}_s, \mathcal{D}_t\}$. The source domain with label information $\mathcal{D}_s = (x_i^s, y_i^s)|_{i=1}^{n_s}$ and the target domain without any label information $\mathcal{D}_t = (x_i^t)|_{i=1}^{n_t}$, where y_i^s is the true label of instance x_i^s in source domain, n_s and n_t are respectively the number of data instances in source and target domains. Our goal is to build a model that can predict target domain correctly.

3.1 The Techniques Used in Step I

We apply Logistical Regression [9] and Naïve Baysian method [19] to select reliable data set from the target domain according to the produced probabilistic predictions.

Logistical Regression. Logistic regression is an approach to learn functions of $P(Y|\mathbf{X})$ in the case where Y is discrete-valued, and $\mathbf{X}$ is any vector containing discrete or continuous random variables. Logistic regression assumes a parametric form for the distribution $P(Y|\mathbf{X})$, then directly estimates its parameters from the training data. The parametric model assumed by logistic regression in the case where Y is Boolean is

$$P(y = \pm 1|\mathbf{x}; \mathbf{w}) = \sigma(y\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + \exp(-y\mathbf{w}^T\mathbf{x})}, \tag{1}$$

where $\mathbf{w}$ is the parameter of the model. Under the principle of *Maximum A-Posteriori* (MAP), $\mathbf{w}$ is estimated under the Laplacian prior. Given a data set $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$, we want to find the parameter $\mathbf{w}$ which maximizes:

$$\sum_{i=1}^{N} \log \frac{1}{1 + \exp(-y_i\mathbf{w}^T\mathbf{x}_i)} - \frac{\lambda}{2}\mathbf{w}^T\mathbf{w}. \tag{2}$$

After **w** is estimated, Equation (1) can be used to compute the probabilities of an instance belonging to the positive and negative classes. Though the Logistical Regression introduced here can only deal with two-class classification problem, it can be naturally extended to tackle multi-class case.

Naïve Baysian. Naïve Baysian is one of popular methods for text classification. Given a set of training documents $\mathcal{D} = (x_i, y_i)|_{i=1}^n$ (n is the number of documents) with m distinct words, each document d_i is considered an ordered list of words, and $x_{i,k}$ denotes the word in position k of x_i. We also have a set of pre-defined classes $C = \{c_1, \cdots, c_l\}$ (l is the number of classes), and need to compute the posterior probability $P(c_j|d_i)$. Based on the Baysian probability and multinomial model, we have

$$P(c_j) = \sum_{i=1}^{n} P(c_j|d_i)/n, \tag{3}$$

and with Laplacian smoothing,

$$P(w_t|c_j) = \frac{1 + \sum_{i=1}^{n} O(w_t, d_i)P(c_j|d_i)}{m + \sum_{s=1}^{m}\sum_{i=1}^{n} O(w_s, d_i)P(c_j|d_i)}, \tag{4}$$

where $O(w_t, d_i)$ is the co-occurrence of word w_t and document d_i, m is the total number of words. Finally, we can compute the posterior probability $P(c_j|d_i)$ under the independent assumption of the probabilities of the words as follows,

$$P(c_j|d_i) = \frac{P(c_j)\prod_{k=1}^{|d_i|} P(x_{i,k}|c_j)}{\sum_{r=1}^{l} P(c_r)\prod_{k=1}^{|d_i|} P(x_{i,k}|c_r)}. \tag{5}$$

The document d_i is predicted as

$$\max_j P(c_j|d_i). \tag{6}$$

When applying the trained model from source domain $\mathcal{D}_s$ to the target domain $\mathcal{D}_t$, we can obtain the resultant prediction probability matrix $A \in R_+^{n_t \times l}$, where n_t is the number of documents in target domain and l is the number of classes, R_+ denotes the set of nonnegative real numbers. Note that we normalize the elements in each row of A, $\sum_{j=1}^{l} A_{i,j} = 1$. Then according to the probability matrix A, the class labels of documents can be predicted by Equation (6). For the example in Table 1, documents $\{d_1, d_2, d_3, d_4, d_5, d_{11}, d_{14}, d_{15}, d_{16}, d_{19}\}$ are predicted as c_1, while documents $\{d_6, d_7, d_8, d_9, d_{10}, d_{12}, d_{13}, d_{17}, d_{18}, d_{20}\}$ are predicted as c_2. Finally, we sort the prediction probabilities of the documents in each class, and select a portion of documents with highest prediction probabilities as reliable data given a selecting rate r. E.g., $r = 0.5$, the selected documents in c_1 are $\{d_1, d_2, d_3, d_4, d_5\}$, and the ones in c_2 are $\{d_6, d_7, d_8, d_9, d_{10}\}$. Note that we select a portion of documents with highest prediction confidence from each class to construct reliable data is trying to ensure that we have "label" data for each class when training a classifier.

3.2 The EM Algorithm in Step II

In this step we develop an EM Algorithm to build a final model which can predict the target domain more correctly. Specifically, we iteratively retrain a model during the

Algorithm 1 Effectively Constructing Reliable Data for Cross-domain Text Classification

Input: Given labeled source domain $\mathcal{D}_s$ and unlabeled target domain $\mathcal{D}_t$. K, the number of iterations. r, the selecting rate.
Output: the model can correctly predict target domain data.

1. Apply the supervised learning algorithm (e.g., NBC or LR) to train a model on source domain, and then predict the target domain $\mathcal{D}_t$ to obtain prediction probability matrix A.
2. Select the reliable data RD based on the prediction probability matrix A and the selecting rate r.
3. Use RD as "labeled" data set to train a model (e.g., NBC), and make a prediction on target domain $\mathcal{D}_t$, we obtain the prediction results $P^{(0)}(c_j|d_i)$.
4. $k := 1$
5. Update $P^{(k)}(c_j)$ and $P^{(k)}(w_t|c_j)$ ($1 \leq j \leq l$, $1 \leq t \leq m$) according to Equations (3) and (4) in *E* step;
6. Update $P^{(k)}(c_j|d_i)$ ($1 \leq i \leq n_t$) according to Equation (5) in *M* step;
7. $k := k + 1$, if $k < K$, turn to Step 5.
8. Output the final model.

EM iterating process according to the prediction results produced by the model in last iteration.

The EM algorithm also contains two steps, the *Expectation* step (*E* step) and the *Maximization* step (*M* step). In our algorithm, the *E* step is to estimate the model, while *M* step is to maximize the posterior probability of target domain data. Our EM algorithm is based on the Naïve Baysian method, so the *E* step corresponds to Equations (3) and (4), and Equation (5) is for *M* step. The description of the proposed two-step method is detailed in Algorithm 1. Note that when the EM algorithm converges, the predicted labels of reliable data may be changed. The reason we update them is that they are not really labeled data, though there is high prediction accuracy on reliable data.

4 Experiments

We conduct experiments on a large amount of classification problems, and focus on binary classification.

4.1 Data Preparation

20Newsgroup[3] is one of the benchmark data sets for text categorization. Since the data set is not originally designed for cross-domain learning, we need to do some data preprocessing. The data set is partitioned evenly cross 20 different newsgroups, and some very related newsgroups are grouped into certain top category. For example, the top category *sci* contains four subcategories *sci.crypt*, *sci.electronics*, *sci.med* and *sci.space*.

We select three top-categories *sci*, *talk* and *rec* (they all have four sub-categories) to perform two-class classification experiments. Any two top categories can be selected to construct two-class classification problems, and we can construct three data sets *sci vs. talk*, *rec vs. sci* and *rec vs. talk* in the experimental setting. For the data set *sci*

[3] http://people.csail.mit.edu/jrennie/20Newsgroups/

vs. talk, we randomly select one subcategory from *sci* and one subcategory from *talk*, which denote the positive and negative data, respectively. The test data set is similarly constructed as the training data set, except that they are from different subcategories. Thus, the constructed classification task is suitable for cross-domain learning due to the facts that 1) the training and test data are from different distributions since they are from different subcategories; 2) they are also related to each other since the positive (negative) instances in the training and test set are from the same top categories. For the data set *sci vs. talk*, we totally construct 144 ($P_4^2 \cdot P_4^2$) classification tasks. The data sets *rec vs. sci* and *rec vs. talk* are constructed similarly with *sci vs. talk*.

4.2 Compared Approaches

In this paper, the supervised learning algorithms Naïve Bayesian (NBC) [19] and Logistical Regression (LR) [9] are used to select the reliable data set, thus we have two methods, called NBC_R_EM and LR_R_EM, respectively. The baseline methods include: 1) Using Naïve Bayesian (NBC) [19] and Logistic Regression (LR) [9] to learn classifiers; 2) Training the classifiers only on the selected reliable data set produced by NBC and LR, respectively denoted as NBC_R and LR_R. 3) The Transductive Support Vector Machine (TSVM) [12] and the state-of-the-art cross-domain classification method (CoCC) [4].

Actually, we can combine and make full use of all the reliable data selected by NBC and LR, and train classifiers on this combination of reliable data[4], denoted as Com_R and Com_R_EM.

The baseline methods LG is implemented by the package[5], and TSVM are given by SVM^{light}[6]. The parameter settings of CoCC is the same as their original paper. The selecting rates $r_1 = 0.2$ for NBC_R_EM and $r_2 = 0.2$ for LR_R_EM, the maximal number of EM iterations is 50. We use the classification accuracy on target domain data to evaluate all compared algorithms.

4.3 Experimental Results

We evaluate all the compared approaches on three date sets *sci vs. talk*, *rec vs. sci* and *rec vs. talk*, and only list the detailed results of *sci vs. talk* due to the space limit. The results are shown in Figure 1, and we have following findings:

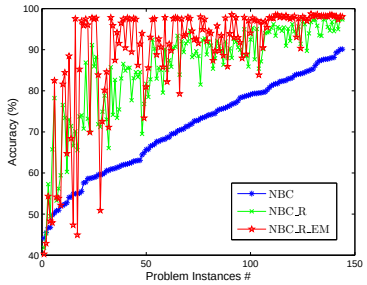
1) In Figure 1(a) and 1(b), the methods NBC_R and LR_R are better than NBC and LR, respectively, which indicates that the reliable data selected from target domain are very effective. Also NBC_R_EM outperforms NBC_R and LR_R_EM outperforms LR_R show the accuracy gains of EM algorithm in the second step. Moreover, NBC_R_EM (LR_R_EM) performs significantly better than NBC (LR), which notes that our proposed methods are more successful to handle cross-domain classification problems.

2) From Figure 1(c), we can find that all the two-step methods outperform Com_R. Also LR_R_EM is better than NBC_R_EM, which may indicate that the reliable data selected by LR is more informative. Though the results of combination Com_R_EM is very similar with LR_R_EM, it seems much more stable. So we can use Com_R_EM as the final

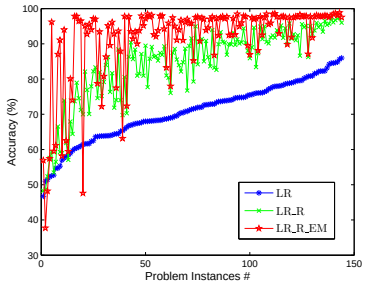
[4] In the combination process, we get rid of the data instances that these two algorithms NBC and LR do not give the same prediction label.

[5] http://research.microsoft.com/∼minka/papers/logreg/

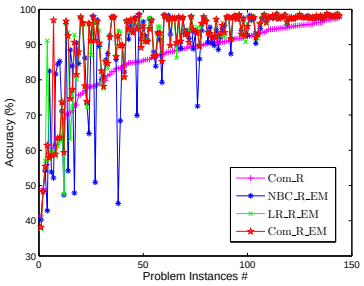
[6] http://svmlight.joachims.org/



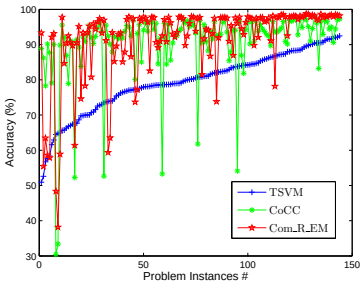
(a) NBC_R_EM vs. NBC_R, NBC



(b) LR_R_EM vs. LR_R, LR



(c) Com_R_EM vs. LR_R_EM, NBC_R_EM, Com_R



(d) Com_R_EM vs. CoCC, TSVM

Fig. 1. The Performance Comparison of All Classification Algorithms on Data Set *sci vs. talk* (The parameters $r_1 = 0.2$, $r_2 = 0.2$)

classifier when we do not know which one of the classifiers NBC_R_EM and LR_R_EM is better.

3) The results in Figure 1(d) show our method Com_R_EM is also superior to TSVM and CoCC.

Furthermore, we record the average performance of all 144 problems from each data set in Table 1, and the best values are marked with bold font. Our methods Com_R_EM, LR_R_EM, NBC_R_EM outperform all the baseline methods, except that on data *set sci vs. talk* CoCC is slightly better than NBC_R_EM. All these results again validate the advantage of the proposed methods.

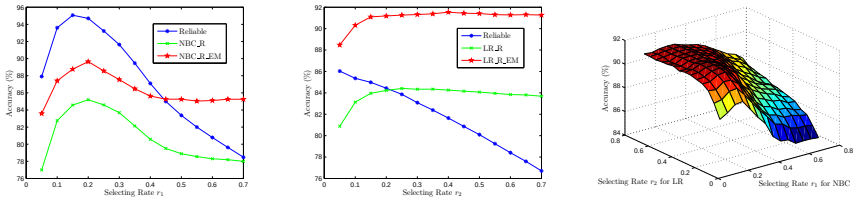
Table 2. Average Performances (%) on 144 Problems for Each Data Set (The parameters $r_1 = 0.2$, $r_2 = 0.2$)

Compared Algorithms		NBC	LR	NBC_R	LR_R	Com_R	TSVM	CoCC	NBC_R_EM	LR_R_EM	Com_R_EM
Data Sets	*sci vs. talk*	70.93	70.64	85.20	84.23	86.00	79.35	90.50	89.65	91.19	**91.39**
	rec vs. sci	68.75	65.57	83.07	80.31	82.43	82.81	87.02	91.09	91.41	**93.21**
	rec vs. talk	74.71	72.49	91.36	91.20	92.87	84.94	93.66	96.44	97.31	**97.33**

4.4 Parameter Affection

We investigate the performance of Com_R_EM affected by the selecting rates r_1 for NBC and r_2 for LR, and sample them in the range of [0.05, 0.7] with an interval 0.05. The results of the average performance over 144 problems under different parameters

are shown in Figure 2. The label "Reliable" in the figures stands for the prediction accuracy on reliable data set. For NBC in Figure 2(a), all the values of "Reliable", NBC_R and NBC_R_EM first increases then decreases with the increasing of selected rate r_1. While for LR in Figure 2(b), the value "Reliable" decreases along with the increasing of r_2, and NBC_R, NBC_R_EM keep stable when r_2 is large enough. From these results, to ensure the good performance of our methods we recommend that the selecting rate should not be set too large or too small, since too small value of selecting rate may not include sufficient information and too large value of selecting rate will introduce more false prediction results when constructing the reliable data set. Also, it is shown again that LR is more stable and much safer to select the reliable data than NBC. In this paper, we set $r_1 = 0.2$ and $r_2 = 0.2$, and Com_R_EM reaches its peak of performance in Figure 2(c).



(a) The Performance Affection of Rate r_1 on Selecting Reliable Data by NBC

(b) The Performance Affection of Rate r_2 on Selecting Reliable Data by LR

(c) The Performance Affection of Rate r_1 and r_2 on Com_R_EM

Fig. 2. The Performance Affection of Rate r_1 and r_2 on Data Set *sci vs. talk*

Running time. Since in each EM iteration, we essentially train a naïve baysian classifier. As you know, the training of naïve baysian classifier is very fast, so our method also can run very fast. Moreover, the EM algorithm can almost converge within 30 iterations on all the classification problems, which shows its efficiency.

4.5 Distribution Mismatch Investigation

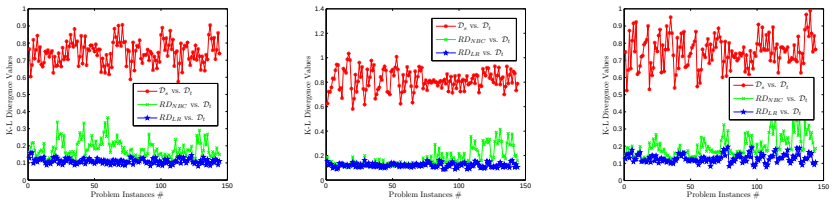
In this subsection we study the distribution mismatch between the source domain and target domain data, and the reliable data and target domain data. K-L divergence [13] is one of the popular evaluation criteria to measure data distribution differences from different domains.

The K-L divergence [13] is computed as follows,

$$K\text{-}L(\mathcal{D}_1||\mathcal{D}_2) = \sum_w P_1(w) \log \frac{P_1(w)}{P_2(w)}, \tag{7}$$

where $P_1(w)$ ($P_2(w)$) is the estimation of word w on $\mathcal{D}_1$ ($\mathcal{D}_2$). If we randomly split the data set from the same domain into training data and test data, then the K-L divergence has a value of nearly zero. The K-L divergence values on three data sets are shown in Figure 3, in which RD_{NBC} and RD_{LR} denote the reliable data set selected by NBC and LR models, respectively. From these results, we can find that the K-L divergence values between the source and target domain data are much larger than the ones between the reliable data and target domain data, which indicates that the selected reliable data

are more similar with the target domain data. This is why the performance of models trained from reliable data are much better than the ones trained from source domain (i.e., see the results in Table 2). Also it can be seen that the K-L divergence values between RD_{LR} and $\mathcal{D}_t$ is smaller and more stable than the one between RD_{NBC} and $\mathcal{D}_t$, which validates the findings in Section 4.3 and 4.4 that the reliable data selected by LR are much more informative and safer.



(a) The Distribution Mismatch on Data Set $scivs.talk$ (b) The Distribution Mismatch on Data Set $recvs.sci$ (c) The Distribution Mismatch on Data Set $recvs.talk$

Fig. 3. The Distribution Mismatch Investigation on Three Data Sets

5 Conclusions

In this paper we investigate the prediction results on target domain predicted by the model learnt from source domain, and find the model can give very good predictions on a certain subset of the target domain data with high prediction confidence. Along this line, we propose a new two-step method for cross-domain learning, in which, we first construct reliable data set with high prediction confidence, and then develop an EM algorithm to build the final classifier. Experimental results show that our methods can handle well the cross-domain classification problems.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 60933004, 60975039, 61175052, 61035003, 61072085), National High-tech R&D Program of China (863 Program) (No.2012AA011003).

References

1. A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proc. of the 11th Annual Conference on Computational Learning Theory*, pages 92–100, 1998.
2. B. E. Boser, I. Guyou, and V. Vapnik. A training algorithm for optimal margin classifiers. In *Proc. of the 5th AWCLT*, 1992.
3. C. C. Chang and C. J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.
4. W. Dai, G. Xue, Q. Yang, and Y. Yu. Co-clustering based classification for out-of-domain documents. In *Proc. of the 13th ACM SIGKDD*, pages 210–219, 2007.
5. W. Dai, Q. Yang, G. Xue, and Y. Yu. Boosting for transfer learning. In *Proc. of the 24th International Conference on Machine Learning (ICML), Corvallis, OR*, pages 193–200, 2007.
6. J. Gao, W. Fan, J. Jiang, and J. W. Han. Knowledge transfer via multiple model local structure mapping. In *Proc. of the 14th ACM SIGKDD*, pages 283–291, 2008.

7. Q. Q. Gu and J. Zhou. Learning the shared subspace for multi-task clustering and transductive transfer classification. In *Proc. of the ICDM*, 2009.
8. J. Z. He, Y. Zhang, X. Li, and Y. Wang. Naive bayes classifier for positive unlabeled learning with uncertainty. In *Proceedings of the 10th SIAM SDM*, pages 361–372, 2010.
9. David Hosmer and Stanley Lemeshow. *Applied Logistic Regression*. Wiley, New York, 2000.
10. J. Jiang and C. X. Zhai. Instance weighting for domain adaptation in nlp. In *Proceedings of the 45th ACL*, pages 264–271, 2007.
11. J. Jiang and C. X. Zhai. A two-stage approach to domain adaptation for statistical classifiers. In *Proceedings of the 16th ACM CIKM*, pages 401–410, 2007.
12. T. Joachims. Transductive inference for text classification using support vector machines. In *Proc. of the 16th ICML*, pages 200–209, 1999.
13. S. Kullback and R.A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.
14. W. S. Lee and B. Liu. Learning with positive and unlabeled examples using weighted logistic regression. In *Proceedings of the 20th ICML*, 2003.
15. D. Lewis and M. Riguette. A comparison of two learning algorithms for text categorization. In *Proc. 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 81–93, 1994.
16. X. L. Li, B. Liu, and S. K. Ng. Negative training data can be harmful to text classification. In *Proceedings of the 2010 Conference on EMNLP*, pages 218–228, 2010.
17. B. Liu, Y. Dai, X. L. Li, W. S. Lee, and P. S. Yu. Building text classifiers using positive and unlabeled examples. In *Proceedings of the 3rd IEEE ICDM*, pages 179–186, 2002.
18. B. Liu, W. S. Lee, P. S. Yu, and X. L. Li. Partially supervised classification of text documents. In *Proceedings of the 19th ICML*, pages 387–394, 2002.
19. A. McCallum and K. Nigam. A comparison of event models for naive bayes text classification. In *Proceedings of the AAAI Workshop on Learning for Text Categorization*, 1998.
20. S. J. Pan, J. T. Kwok, and Q. Yang. Transfer learning via dimensionality reduction. In *Proceedings of the 23rd AAAI*, pages 677–682, 2008.
21. S. J. Pan, X. C. Ni, J. T. S, Q. Yang, and Z. Chen. Cross-domain sentiment classification via spectral feature alignment. In *Proceedings of the 19th WWW*, pages 751–760, 2010.
22. S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang. Domain adaptation via transfer component analysis. In *Proceedings of the 21st IJCAI*, pages 1187–1192, 2009.
23. Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE TKDE*, 22(10):1345–1359, 2010.
24. J. Rocchio. Relevance feedback in information retrieval. *The SMART Retrieval System*, pages 313–323, 1971.
25. U. Selen and C. Jaime. Feature selection for transfer learning. In *Machine Learning and Knowledge Discovery in Databases*, volume 6913, pages 430–442. Springer Berlin / Heidelberg, 2011.
26. B. Zadrozy. Learning and evaluating classifiers under sample selection bias. In *Proceedings of the 21th ICML*, pages 114–121, 2004.
27. B. Z. Zhang and W. L. Zuo. Learning from positive and unlabeled examples: A survey. In *Proceedings of ISIP*, pages 650–654, 2008.
28. Y. Zhen and C. Q. Li. Cross-domain knowledge transfer using semi-supervised classification. In *Proceedings of the 21st AJCAI*, pages 362–371, 2008.
29. F. Z. Zhuang, P. Luo, H. Xiong, Q. He, Y. H. Xiong, and Z. Z. Shi. Exploiting associations between word clusters and document classes for cross-domain text categorization. In *Proc. of the SIAM SDM*, pages 13–24, 2010.
30. F. Z. Zhuang, P. Luo, H. Xiong, Y. H. Xiong, Q. He, and Z. Z. Shi. Cross-domain learning from multiple sources: A consensus regularization perspective. *IEEE TKDE*, pages 1664–1678, 2010.