Predictive Distribution Matching SVM for Multi-domain Learning

Chun-Wei Seah[1], Ivor W. Tsang[1], Yew-Soon Ong[1], and Kee-Khoon Lee[2]

[1] School of Computer Engineering, Nanyang Technological University, Singapore
{Seah0116,IvorTsang,asYSOng}@ntu.edu.sg
[2] Institute of High Performance Computing, Singapore
leekk@ihpc.a-star.edu.sg

Abstract. Domain adaptation (DA) using labeled data from related source domains comes in handy when the labeled patterns of a target domain are scarce. Nevertheless, it is worth noting that when the predictive distribution $P(y|\mathbf{x})$ of the domains differs, which establishes *Negative Transfer* [19], DA approaches generally fail to perform well. Taking this cue, the Predictive Distribution Matching SVM (PDM-SVM) is proposed to learn a robust classifier in the target domain (referred to as the target classifier) by leveraging the labeled data from only the relevant regions of multiple sources. In particular, a k-nearest neighbor graph is iteratively constructed to identify the regions of relevant source labeled data where the predictive distribution maximally aligns with that of the target data. Predictive distribution matching regularization is then introduced to leverage these relevant source labeled data for training the target classifier. In addition, progressive transduction is adopted to infer the label of target unlabeled data for estimating the predictive distribution of the target domain. Finally, extensive experiments are conducted to illustrate the impact of Negative Transfer on several existing state-of-the-art DA methods, and demonstrate the improved performance efficacy of our proposed PDM-SVM on the commonly used multi-domain Sentiment and Reuters datasets.

Keywords: Domain Adaptation, Negative Transfer, Predictive Distribution Matching, Progressive Transduction.

1 Introduction

Sentiment classification is an important task [3] for the marketer to predict sentiment polarity (e.g. positive or negative) of user reviews collected for different products. For instance, there are different categories of products from Amazon: books, DVDs, electronics and kitchen appliances. Users' comments of these products are usually described by some common words. Traditional machine learning algorithms can be used to train a sentiment classifier from manually labeled feedbacks for each of these reviews. When a category of products does not have much labeled reviews (referred to as target domain), Domain Adaptation (DA) methods come into hand as these methods can leverage labeled reviews from some

J.L. Balcázar et al. (Eds.): ECML PKDD 2010, Part I, LNAI 6321, pp. 231–247, 2010.

related products referred to as source domains. Besides sentiment classification, DA methods have also been applied in many real applications ranging from Natural Language Processing [13,4,9], text categorization [17], visual concept detection [11,10], WiFi localization [17] and remote sensing [5].

One of the major challenges of leveraging source domains to train a target classifier lies on the dissimilarity of predictive distribution among different domains which will be illustrated by an example in Section 3.2. However, many works on DA are assuming that the predictive distribution between target and source domains is the same [12,8,16,20]. However, Bruzzone and Marconcini [5] explained that when there are limited labeled data, these labeled data do not represent the general population especially for imbalance problem, and introduce a bias in estimating the predictive distribution (e.g. by Naïve Bayes Classifier). In most cases, the target domain has very few labeled data and source domains might have different class distribution from the target one. Thus, this might easily lead to dissimilarity of the predictive distribution between the domains.

In addition, the true class distribution of the target domain is unknown as the labeled data are limited, therefore re-sampling strategies (e.g. SMOTE [6]) for adjusting the source domain to have the same class distribution with the target domain might not be directly applicable in this setting.

Since the predictive distribution of the source domains might differ from the target domain, the classifier directly trained with all labeled data from multiple source domains might not classify well on unlabeled data in the target domain. Direct transferring of knowledge from the source domains to the target domain may also lead to adverse effects, which is referred to as *negative transfer* [19]. In this work, we propose a novel DA method, namely Predictive Distribution Matching Support Vector Machine (PDM-SVM), to address the challenges arisen from the difference in predictive distribution among multiple domains.

The main contributions in this paper are as follows: 1) A k-nearest neighbor graph is iteratively constructed to identify relevant source labeled data that have high similarity in predictive distribution of the target data. Then we exploit this dependency to define the so-called predictive distribution matching regularization that leverages only relevant source labeled patterns to train the target classifier. 2) We demonstrate how to infer the pseudo-labels of target unlabeled patterns by the use of progressive transduction which eventually learns the predictive distribution of the target domain. 3) We illustrate how the negative transfer affects SVMs trained with source domains, Semi-Supervised Learning (SSL) and DA methods when the source and target domains have dissimilar class distribution in Section 3.2. We show that our PDM-SVM approach can handle this problem and significantly outperform those methods in the comprehensive experiments on Sentiment and Reuters datasets.

2 Related Works

Initial work of leveraging labeled patterns from a source domain for the target domain was proposed to minimize the weighted empirical risk for both the source

and target domains [21], which does not consider the distribution difference between the two domains. To address this, several instance weighting methods [13] had been proposed, but these methods usually require considerable amount of target labeled data in order to robustly re-weighting the training instances.

However, target labeled patterns are scarce. Instead of using many target labels, several methods [4,9,17,18] had been proposed to extract some useful features to be augmented in the original feature space. For example, a heuristic method was proposed in [4] to identify some *pivot features* representing common feature structure between different domains to learn an embedded space. Then this space is augmented to the original input feature vector for domain adaptation. Another example is Feature Augmentation (FA) [9], which augments features belonging to the same domain by twice that of the original features so that data within the same domains would be treated as more similar than data in different domains.

Recent DA works [8,16,20,10] are taking up the challenge of learning from multiple source domains. Crammer *et al.* [8] assumed that the distribution of multiple sources is the same, and the change of output labels is a result of varying noise. Luo *et al.* [16] maximized the consensus of predictions from multiple sources. In [20], the authors proposed a Multiple Convex Combination of SVM (M-SVM) trained from multiple source domains and a target domain. However, some source domains may not be useful for knowledge transfer. In [10], a domain-dependent regularizer was proposed to enforce that the prediction of the target classifier on target unlabeled data is close to the prediction of source classifiers from similar sources only. Recently, Domain Adaptation SVM (DASVM) [5] was proposed to tackle the mismatch of the predictive distribution between the domains by removing all source labeled patterns progressively; meanwhile, using Progressive Transductive SVM(PTSVM) [7] to infer the label of target unlabeled patterns by using all remaining source and target labeled data. However, when there are many overlapping sources and target data, and the label of some source labeled data are not consistent with the label of the target data, all these methods might not perform well.

DA is also similar to several learning paradigms such as multi-task learning and multi-view learning. The major difference between DA and multi-task learning is that DA learns a classifier for a specific task in the target domain whereas multi-task simultaneously learns for multiple tasks. For multi-view learning, a classifier is trained for each source domain and labels the unlabeled data when all these source classifiers agree on the predicted output of the unlabeled data to a certain degree, thus assuming the source domains have the same distribution.

3 Predictive Distribution Matching SVM

3.1 Preliminaries and Problem Statements

Throughout the rest of this paper, whenever superscript ^{s} and ^{t} appear in the contents, they represent source domain and target domain respectively. A summary of all important symbols can be found in Table 1.

Table 1. Symbol Definition

Symbol	Definition
m	Total number of domains, the first (m-1) domains represent source domains and the last domain, mth domain, is the target domain
$\mathbf{x}$	Feature vector of a data
y	Class Label for the data x or pseudo-label which is a class label that can be learned for a particular $\mathbf{x}$ that is described in Section 3.5,
n_r	Number of labeled data in rth domain. For the target domain, it is the combination of labeled and pseudo-labeled data
n	$\sum_{r=1}^{m} n_r$
D_L^s	$\cup_{r=1}^{m-1}\{\mathbf{x}_i^r, y_i^r\}^{n_r}$, all labeled data in all source domains
D_L^t	$\{\mathbf{x}_i, y_i\}^{n_m}$, all labeled data in target domain
$\mathcal{D}_L$	$D_L^s \cup D_L^t$
$\mathbf{x}_i^u$	Feature vector of ith unlabeled data in target domain
D_U	All unlabeled data in target domain
$P(\mathbf{x})$	Marginal distribution
$P(y\|\mathbf{x})$	Predictive distribution

Recall that labeled patterns of one domain can be drawn from the joint distribution $P(\mathbf{x}, y) = P(y|\mathbf{x})P(\mathbf{x})$. We also let $P(\mathbf{x})$ be the marginal distribution of the input sets $\{\mathbf{x}_i\}^{n_r}$ in the rth domains. DA methods usually assume that $P^t(\mathbf{x})$ of the target domain and $P^s(\mathbf{x})$ of the source domain are different. Then the task of DA is to predict the labels y_i^t's corresponding to the inputs $\mathbf{x}_i^t$'s in the target domain. Notice that DA is different from Semi-Supervised Learning (SSL). SSL methods employ both labeled and unlabeled data to achieve better prediction performance, in which the labeled and unlabeled data are usually assumed to be drawn from the same domain. It is also worth noting that the common assumption in many DA methods [12,8,16,20] is that $P^s(\mathbf{x}) \neq P^t(\mathbf{x})$, but the source and target domains share the same predictive distribution, *i.e.*, $P^s(y|\mathbf{x}) = P^t(y|\mathbf{x})$, where $P^s(y|\mathbf{x})$ and $P^t(y|\mathbf{x})$ are the predictive distribution of the source and target domains, respectively. This is also referred to as covariate shift [2]. Hence in this work, we attempt to solve domain adaptation in the setting where the predictive distribution is not to be preserved, i.e. $P^s(y|\mathbf{x}) \neq P^t(y|\mathbf{x})$. This can be materializing by diverse class distribution and limited samples in each domain, and by class label inconsistency among different domains.

3.2 An Illustrating Example

Before we introduce our proposed method, in this subsection, we first study how the dissimilarity of the class distribution between the target and source domains affects Domain Adaptation (DA) and Semi-Supervised Learning (SSL) methods. Suppose that there are very few labeled data but a lot of unlabeled data in the target domain, we vary Positive Class Ratio (PCR) of the source domains. Here, PCR defines the percentage of positive class data in the source domains. For example, PCR = 0.1 implies that 10% of the data are positive. Note that when PCR is skewed towards either extremes, the class distribution of the source domains becomes very imbalanced. As mentioned in Section 1, when the data is imbalanced, the limited labeled data might introduce a bias in estimating the predictive distribution. Thus, the difference in the predictive distribution between the target and source domains might occur. To study this,

we train two SVM models: SVM_T trained with only labeled data $\mathcal{D}_L^t$ from target domain; SVM_S trained with only labeled data $\mathcal{D}_L^s$ from all source domains. We include two SSL methods: Transduction SVM (TSVM) [15], trained with the labeled data from all the source and target domains, $\mathcal{D}_L$, and unlabeled data in target domain, $\mathcal{D}_U$; LapSVM [1], the training set is the same as TSVM. We also compare with a DA algorithm: SVM_ST [21], is a SVM trained with $\mathcal{D}_L$.

Here, we will demonstrate the trends of different SVM-based algorithms according to different PCR settings on Sentiment dataset. The task is to classify whether the reviews are positive or negative. The target domain is the reviews of *Electronics* while the source domains are the reviews of *Book*, *DVDs* and *Kitchen appliances*. In the present setup, the test set is designed to contain equal amount of positive and negative data. The source domains however possess different PCR values. The details of other experimental setup will be described in Section 4.

Testing accuracy of different methods against varying PCR in the source domains are reported in Figure 1. Firstly, it is worth to observe that by using both the source and target data, the TSVM, LapSVM and SVM_ST can perform better than SVM_S and SVM_T which use only source and target data respectively. Secondly, one can observe that most methods perform optimally for PCR of 0.5 (i.e. the source and target domains have the same ratio of positive and negative data). However, the performances of all methods dropped sharply when the PCR is skewed toward either extremes (i.e. the source domains are very imbalanced, or the source and target domains have different class distribution), which implies $P^s(y|\mathbf{x})$ and $P^t(y|\mathbf{x})$ would most likely be dissimilar [5]. In addition, leveraging source labeled data from other domains lead to adverse effects on domain adaptation, which is regarded as negative transfer [19]. It is clear that those values below the line of SVM_T can be indicated as negative transfer, as the accuracy of a classifier borrowing labeled data from other source domains performs worse than just using the available target labeled data. The possible reason is that the source and target domains have different predictive distribution, when the source and target domains are combined together as a training set, which represents another predictive distribution and does not reflect the true population of its own domain. Therefore, all classifiers trained with this training set might have poorer generalization performance.

Interestingly, it can be observed that most of the values reported for the SSL methods are above the SVM_T value. A possible reason might be these two methods use target unlabeled data as the regularization. TSVM enforces unlabeled data to have same class ratio as labeled dataset. Therefore when PCR is 0.5, TSVM will classify the unlabeled data into half of them as positive and another half as negative which is the true class distribution for the unlabeled data. Hence this might cause it to perform the best when PCR setting is 0.5. In other PCR settings, TSVM classifies the unlabeled data into the same class ratio as the PCR setting, and suffers poorer classification performance.

Note, LapSVM assumes that if two patterns are close together in high density region, these patterns should have similar predictive outputs. If the manifold assumption holds strongly, LapSVM should perform well in all PCR settings.

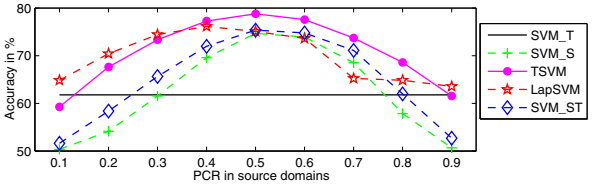

Fig. 1. Testing accuracies on Sentiment Data with varying PCR in source domains and *Electronics* as target domain

However, from our experiences, this is not the case on Sentiment dataset because two reviews with similar comments can have different meanings. For example, "I really like this" and "I dare you would really like this". For the former sentence, it is a positive feedback whereas the latter sentence is a totally negative comment. Thus, LapSVM achieves lower accuracy than just using supervised labeled information (i.e. SVM_ST) in some PCR settings. We also observe that LapSVM performs better than TSVM at both extreme ends of PCR settings, it is possibly because manifold regularization [1] on imbalanced data might be more robust than *cluster assumption* in TSVM [15].

3.3 Predictive Distribution Matching across Multiple Domains

From all the observations in Section 3.2, we are motivated to introduce a new DA method by using target unlabeled data and explicitly considering the predictive distribution of both the source and target data for multi-domain learning. Here, we define a regularizer such that two similar patterns $\mathbf{x}_i^r$ and $\mathbf{x}_j^c$ from the rth and cth domains respectively would produce similar predictive outputs for a *positive transfer* (i.e. two patterns have a high relevance measured by W_{ij}^{rc}):

$$\Omega(f) = \frac{1}{n^2} \sum_{r,c=1}^{m} \sum_{i=1}^{n_r} \sum_{j=1}^{n_c} (f(\mathbf{x}_i^r) - f(\mathbf{x}_j^c))^2 W_{ij}^{rc}, \tag{1}$$

where n_r and n_c are the number of patterns in the rth and cth domains respectively. Here, $\mathbf{x}_i^r$ is the ith data in the rth domain and $\mathbf{x}_j^c$ is the jth data in the cth domain. The similarity W_{ij}^{rc} to measure a *positive transfer* of two patterns $\mathbf{x}_i^r$ and $\mathbf{x}_j^c$ is defined as follows:

$$W_{ij}^{rc} = \sum_{z=1}^{v} P^r(y_z|\mathbf{x}_i^r) P^c(y_z|\mathbf{x}_j^c) I[y_i = y_j] D[r \neq c] S(\mathbf{x}_i^r, \mathbf{x}_j^c), \tag{2}$$

where v is the number of classes, $P^r(y|\mathbf{x}_i^r)$ is the predictive distribution of the rth domain on pattern $\mathbf{x}_i^r$ which can be estimated by means of Naïve Bayes Classifier on labeled data, while $I(\cdot)$ and $D(\cdot)$ are indicator functions. Here, $S(\mathbf{x}_i^r, \mathbf{x}_j^c)$ measures the similarity between patterns $\mathbf{x}_i^r$ and $\mathbf{x}_j^c$, and is defined as the weight of an edge in a graph constructed by k nearest neighbors.

In this work, we also define the *predictive distribution matching score* for two nearby patterns $\mathbf{x}_i^r$ and $\mathbf{x}_j^c$ as $\sum_{z=1}^{v} P^r(y_z|\mathbf{x}_i^r) P^c(y_z|\mathbf{x}_j^c)$ for measuring the similarity of the predictive distribution of two patterns, where $\sum_{z=1}^{v} P^r(y_z|\mathbf{x}_i^r)$ and

$\sum_{z=1}^{v} P^c(y_z|\mathbf{x}_j^c)$ are both equal to 1. Moreover, those patterns not in the same class will be disconnected, as the indicator function $I[y_i = y_j]$ returns a logic of 1 if both labels are the same, otherwise it returns 0. Intuitively, this indication function can be viewed as a pairwise constraint that two patterns in the same class can be linked together whereas two patterns belonging to different class should not be linked [22]. As we do not assume manifold assumption in each domain, we use an indicator function $D[r \neq c]$ to allow only data in different domains to be connected. Note that if the target domain follows manifold assumption, the manifold regularizer can be easily added into our formulation. But in this paper, we do not assume manifold property in any dataset and hence our method can apply in general cases.

From the definition of W_{ij}^{rc} in (2), two similar patterns from different domains having a high response of the predictive distribution matching score and the same class label would share similar predictive outputs. Therefore, we can identify relevant source labeled data from the data having high similar predictive distribution for domain adaptation.

3.4 Proposed Formulation

In our regularization framework, the decision function is learned by minimizing the following regularized risk functional: $f^* = \arg\min_f \gamma_A \|f\|^2 + \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f(x_i))$, where $\ell(y_i, f(x_i))$ denotes the loss function and γ_A controls the smoothness of the solutions. In this paper, we employ hinge loss function of SVM as $\ell(\cdot)$. Together with our proposed data-dependent regularizer in (1), the regulated risk functional for domain adaptation is then formulated as:

$$\min_f \gamma_I \Omega(f) + \gamma_A ||f||^2 + \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f(x_i)), \tag{3}$$

where γ_I regulates the decision function $f(\mathbf{x})$ according to our proposed regularizer (1) for multiple source domain adaptation. We refer our proposed method to as Predictive Distribution Matching SVM (PDM-SVM). Note that our regularizer can be easily added into other standard regularization frameworks. We use SVM as our formulation since we are investigating and comparing with other SVM-based methods.

By defining a Laplacian matrix $L = D - W$ where W is a $n \times n$ matrix with entries defined in (2) and D is a $n \times n$ diagonal matrix with diagonal entry $D_{ii} = \sum_{j=1}^{n} W_{ij}$, the resultant optimization problem (3) can be formulated as LapSVM formulation [1]. Thus allowing us to take advantage of existing LapSVM algorithm to solve (3) by using (2) as the Laplacian matrix L in the algorithm. By duality, the minimization problem (3) is equivalent to the dual problem:

$$\max_{\alpha \in \Re^n} \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\alpha' Y K \left(2\gamma_A I + \frac{2\gamma_I}{n^2} L K\right)^{-1} Y \alpha, \tag{4}$$

where $Y = diag(y_1, \ldots, y_n) \in \Re^{(n \times n)}$, K is the $n \times n$ kernel matrix and I is an identity matrix . The decision function is defined as follows:

$$f(x) = \sum_{i=1}^{n} \beta_i K(x, x_i), \tag{5}$$

where $\beta = (2\gamma_A I + 2\frac{\gamma_I}{n^2}LK)^{-1}Y\alpha$. For more details of the derivations, interested reader may refer to [1].

3.5 Progressive Transduction on $\mathcal{D}_U$

One of major challenges of domain adaptation is that the prediction distribution $P^t(y|\mathbf{x})$ cannot be well-estimated with limited labeled data in the target domain. Therefore, many existing DA algorithms [12,17] assume that the target domain and the sources domains share the same prediction distribution, i.e., $P^s(y|\mathbf{x}) = P^t(y|\mathbf{x})$. However, as illustrated in Figure 1, when the predictive distribution varies across domains, DA and SSL methods may have impaired performance. In this subsection, we propose to use progressive transduction method for acquiring the additional labeled data to estimate $P^t(y|\mathbf{x})$.

Progressive transduction is to progressively label certain number of unlabeled data with pseudo-label which are the most confident predicted outputs in current iteration. These learned pseudo-labeled data are then used to estimate the predictive distribution $P^t(y|\mathbf{x})$. After that, we apply the learned $P^t(y|\mathbf{x})$ to our proposed regularizer (1) for multiple source domains adaptation and train a new classifier with the newly added pseudo-labeled data using (4). The progressive transduction step is then repeated until it reaches the stopping criterion.

In jth iteration, a classifier is trained using the available labeled and pseudo-labeled data using (4). Then the classifier predicts the unlabeled data using decision function (5). Let us group these unlabeled data into their predicted classes and assign them with labels accordingly before sorting the positive set in decreasing order and negative set in increasing order as follows:

$$T_+^j = \{(x_i^u, +)|x_i^u \in D_u^j, f^j(x_i^u) \geq f^j(x_{i+1}^u) \geq 0\}, \tag{6}$$

$$T_-^j = \{(x_i^u, -)|x_i^u \in D_u^j, f^j(x_i^u) \leq f^j(x_{i+1}^u) < 0\}, \tag{7}$$

where $D_u^j = D_u \backslash B^{j-1}$ and B^j are all the pseudo-labeled data from the start of initialization to the current jth iteration, which is defined as follows:

$$B^j = B^{j-1} \cup B_+^j \cup B_-^j, \tag{8}$$

where B^{j-1} is all the pseudo-labeled data from the start of initialization to the $(j-1)$th iteration and the current jth iteration's pseudo-labeled data are from B_+^j and B_-^j which are defined as follows:

$$B_+^j = \{(x_i^u, y_i^u) \in T_+^j | 1 \leq i \leq P_+^j\}, \tag{9}$$

$$B_-^j = \{(x_i^u, y_i^u) \in T_-^j | 1 \leq i \leq P_-^j\}, \tag{10}$$

where $P_+^j = min(p, |T_+^j|)$, $P_-^j = min(p, |T_-^j|)$, p is a hyper-parameter. Hence, the pseudo-labeled set B_+^j contains data with the highest p predictive values in T_+^j and B_-^j contains data with the lowest p predictive values in T_-^j. Therefore, the pseudo-labeled data learned from the current iteration are the most confident predicted labels as they are the furthest away from the decision boundary. Then these pseudo-labeled patterns are being incorporated as part of training set in (4). As these pseudo-labeled data and the target labeled domain are from

Algorithm 1(PDM-SVM)
1. Initialize $B^0 = \emptyset$,$F^t = \emptyset$, M=m% of unlabeled data
2. While $\|B^{j-1}\| < M$
3. Build B^j using (8)
4. Train the classifier F^t in (4) using D_L and B^j
5. if $\|D_L\| + \|B^j\| >= \Theta$
6. Train a classifier F^t in (4) using D_L^t and B^j

the same domain, their predictive distribution $P^t(y|\mathbf{x})$ is re-estimated to compute W_{ij}^{rc} in (2) for the target domain in each iteration. When m% of the entire unlabeled data are incorporated as part of the training set, the whole progressive transduction process terminates. After that, if the size of the combination of target labeled dataset and pseudo-labeled dataset is larger than a certain threshold Θ, then SVM is trained using only the target labeled data and the pseudo-labeled patterns so the final classifier would consist only target data to represent the true distribution for the target data. The detailed algorithm of PDM-SVM is presented in Algorithm 1. Finally, the final trained classifier is used to classify the test data using decision function (5).

3.6 Demonstration of PDM-SVM on a Synthetic Dataset

Besides the dissimilarity of class distribution among multi-domains, here we also consider the class label inconsistence from multi-domains, where each domain has its own modality. We generate a synthetic dataset (Figure 2) to depict how PDM-SVM uses the predictive distribution similarities between two patterns from different domains to construct a graph. In this dataset, there are three labeled source domains and a target domain that contain only unlabeled data.

Figure 2(a) depicts the target domain. Figure 2(b) shows the first two source domains having their positive and negative data overlapping with the target positive and negative data respectively and the third source domain having its negative data overlapping with the target negative data, but its positive data are near to the target negative data. Intuitively, those target data close to the positive data of the third source will be classified as positive, which is undesirable. However, this is the case for SVM_ST and its decision boundary is depicted in Figure 2(c) by a thinner curve line. As the entire dataset is formed from several domains, and each domain has its own modality, the dataset could become a multi-modality problem. Hence, traditional DA methods which cannot handle multi-modal datasets would fail as the classifier trained with all labeled data has incorrect decision boundary. Whereas, PDM-SVM can classify all the target data correctly, and its decision boundary is shown by the thicker curve line. This is because PDM-SVM can iteratively construct a graph to identify relevant source data as shown in Figure 2(d).

Figure 2 (d) shows that PDM-SVM can construct a graph using the predictive distribution similarity between two patterns from different domains. The lower rectangular box depicts the connections of one target pseudo-labeled learned by PDM-SVM with several source data points which demonstrates that PDM-SVM

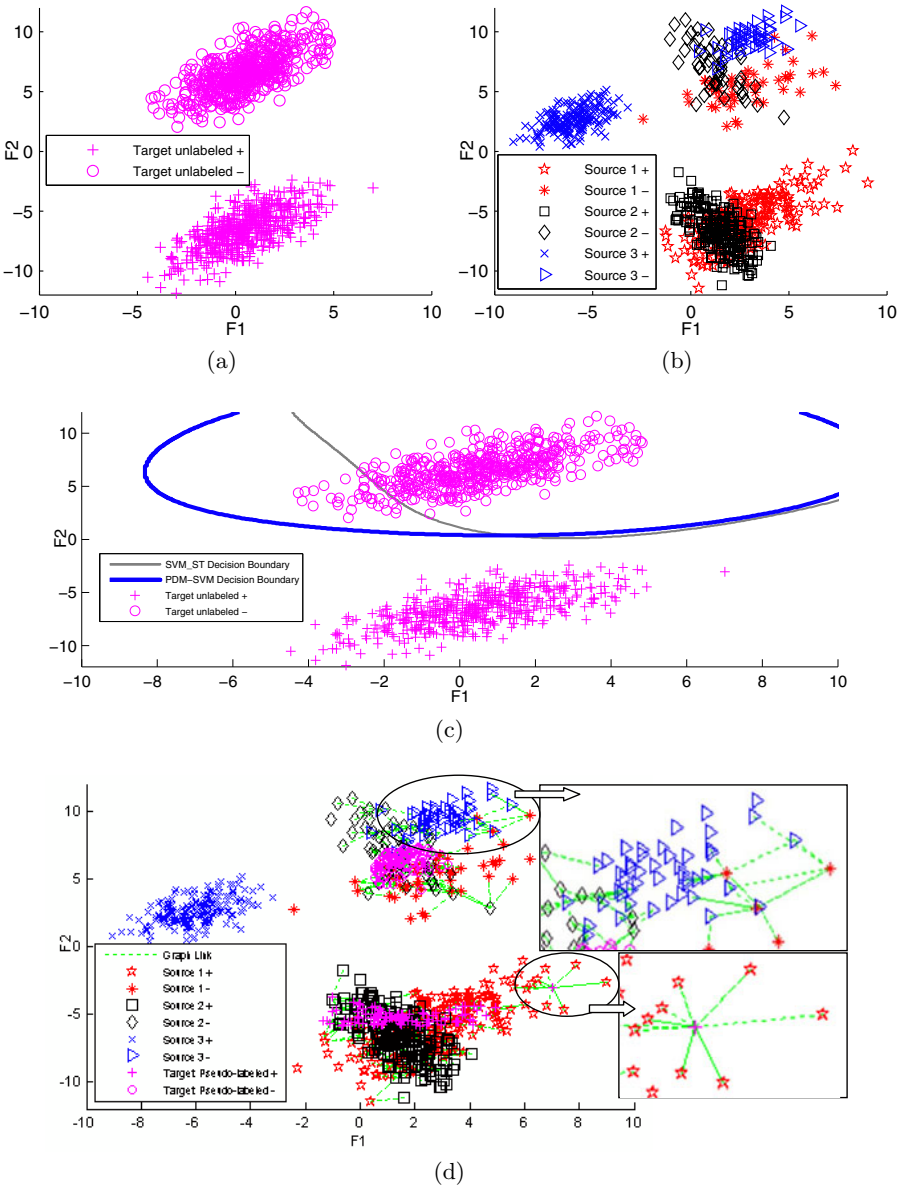

Fig. 2. PDM-SVM demonstration using a Synthetic dataset. The data are in 2-Dimension represented by two features, F1 and F2 respectively.
(a) Target domain with 500 positive and 500 negative instances. (b)Three source domains related to the target domain except the positive data of the third source domain near to the target negative data. Each domain consists of 150 positive and 50 negative instances (c) The decision boundary of SVM_ST and PDM-SVM. The space below the decision boundary is classified as positive, whereas the other side is classified as negative. (d)The graph's connections between labeled and pseudo-labeled data which are learned by PDM-SVM in the final iteration.

can identify pseudo-labeled data from the constructed graph. The upper rectangular box depicts negative data mainly in the third domain where data closer to other data in different domains are connected and data that are further apart from the data in other domains are not connected. When certain regions having many nodes with high predictive distribution values are connected together, these regions can be used to reflect the predictive output of the target regions. Hence, the pseudo-labeled data can be learned from these regions and eventually PDM-SVM can learn a classifier using these pseudo-labeled data.

4 Experiments

In this Section, we investigate several existing state-of-the-art SVM-based methods, DA methods and the proposed PDM-SVM under a multi-domain setting of differing predictive distribution and scare target labels. SSL methods are also considered in the present study to see how they perform when they use target unlabeled data as part of their training. Note that DA methods (e.g. [13]) that require considerable number of target labels to function and cater only to single source domain are omitted. Here, apart from investigating the methods considered in Section 3.2, we also include additional DA learning algorithms such as: M-SVM, [20], is a linear combination of SVMs trained with $\mathcal{D}_L^s$'s and $\mathcal{D}_L^t$; FA, [9], is trained with $\mathcal{D}_L$; DASVM, [5], is trained with $\mathcal{D}_L$ and $\mathcal{D}_U$.

4.1 Experimental Setup

The parameters of the DA methods are configured by means of k-fold cross-source domains validation as suggested in [14] (an extension of k-fold cross validation for domain adaptation) and are tabulated in Table 2. For methods using only labeled data, i.e. SVM_S, SVM_B, SVM_ST, M-SVM and FA, each partition represents a source domain in k-fold cross-source domains validation. For methods using both labeled and unlabeled data, i.e. TSVM, LapSVM, DASVM and PDM-SVM, the kth source domain is used as the labeled data in the kth fold evaluation, while the rest are used as unlabeled data and for validation.

Table 2. Parameter Settings

Classifiers	Parameter Settings
SVM_S, TSVM, SVM_ST, M-SVM, FA & DASVM	C is chosen by cross-validation.
LapSVM & PDM-SVM	γ_A and γ_I are chosen by cross-validation. Using 6 nearest neighbors and normalized Laplacian matrix to construct the graph. The weight for Laplacian matrix is based on cosine distances, as commonly used in text classification.
SVM_T & SVM_B	C is fixed as 1 since the labeled data are limited. For example, in Reuters dataset setting, the target domain will only consist two labeled data.
Other parameters	p in DASVM and PDM-SVM is fixed as 5. β is fixed as 3.10^{-2} for DASVM which is the same value used in [5]. m and Θ for PDM-SVM are set as 20% and 50, respectively.

4.2 Datasets

In the present study, we consider two datasets namely Sentiment and Reuters-21578. Sentiment is a popular multi-domain benchmark dataset, defined in [3]. It is typically used in the context of DA and consisted of even positive and negative class distribution, hence it is used here to synthesize diverse PCR settings for investigating the robustness of SSL and DA methods. Reuters dataset, on the other hand, allows us to study the efficacy of SSL and DA methods in the presence of uneven class distribution in each domain.

In the experimental study, we further pre-process the datasets by extracting only the single-terms, removing all stopwords, normalizing each feature and performing stemming. Finally, each feature of a review is represented by its respective *tf-idf* value, and linear kernel is used in the experiments.

Multi-Domain Sentiment Dataset. It is generated from *Amazon.com* comprising four categories of product reviews: *Book*, *DVDs*, *Electronics* and *Kitchen appliances*. Each category of product review is considered as a domain and comprises of 1000 positive and 1000 negative reviews. For each task, we used one dataset as target domain while the rest as related source domains. For a target domain, we randomly selected 10 positive and 10 negative instances as labeled data and keeping the rest as unlabeled data. In regards to each source domain, we randomly selected 200 to form the labeled data. To study the mismatch in predictive distribution between the source and target domains, 9 different PCR settings are generated for investigations. The 9 PCR settings are chosen from 0.1 to 0.9 at an incremental of 0.1 step size. For example, in a setting of PCR of 0.1, out of the 200 data selected for each source domain, 20 positive data are selected while the rest make up the negative data. To study the performance of an ideal SVM_ST with prior knowledge on class distribution, we consider here additional SVM_B classifier. For each source domain, SVM_B re-samples the data to have the same PCR as the target domain. Let ρ and η be the number of positive and negative samples in each source domain respectively. Since our target unlabeled data has equal number of positive and negative samples, then for each PCR setting, the classifier re-samples both positive and negative samples as $min(\rho, \eta)$ in each source domain. Thus all domains have the same class distribution.

Multi-Domain Reuters Dataset. 3 out of 4 main categories of the dataset namely *People*, *Organizations* and *Exchanges* are considered in the present study, thus resulting 3 tasks being experimented: *People* versus *Organizations*, *People* versus *Exchanges* and *Organizations* versus *Exchanges*. *Places* category is not used due to the vast instances belonging to this category that overwhelms all other categories, thus making the study fruitless. Further, in each main category, the subcategory with largest dataset is used as target domain while the remaining 4 largest subcategories as related source domains. Then in each task, the xth largest subcategory of a main category is labeled as positive while the xth largest subcategory from another main category is labeled as negative. All data in the source domains are used as labeled data and for the target domain, one positive and one negative data are randomly selected to form the labeled

data while the rest are used as unlabeled data. Note that this dataset has uneven positive and negative samples in each subcategory, hence the testing distribution is imbalanced and the predictive distribution of the source domains is quite diverse with respect to one another. Furthermore, since it is not always feasible to re-sample the source domains to match the SVM_B setting, it is not considered in the study of this dataset.

4.3 Results and Discussions

We first study the performance of various classifiers with varying PCR in the source domains on Sentiment dataset. For the sake of conciseness, Figure 3 depicts the testing accuracies on Sentiment data for 9 different PCR settings in the source domains, with Electronics and Kitchen Appliances as the target domain. Note that PCR of the target unlabeled dataset is confirmed at approximately 0.5, hence when the PCR of the source domains is also in the region of 0.5, the predictive distribution of the source domains is most likely to be similar to the target unlabeled dataset. The rest of the PCR settings on the other hand would likely result in mismatch of predictive distribution between the source and target domains. Each of the four domains in the Sentiment dataset will take turns to be used as the target domain. Their detailed results for PCR at 0.3, 0.5 and 0.7 are then reported in Table 3.

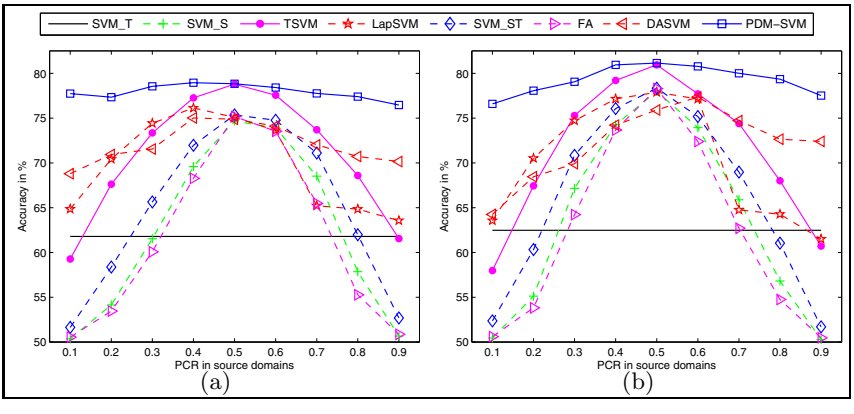

Fig. 3. Testing accuracies on Sentiment dataset for varying PCR in source domains. (a) Target domain is Electronics (b) Target domain is Kitchen Appliances.

As shown in Figure 3, at both extreme ends of the PCR settings, for the same labeled and unlabeled data, LapSVM and TSVM are found to underperform DASVM and PDM-SVM. This is expected since LapSVM and TSVM do not consider the predictive distribution mismatch between different domains. Furthermore, TSVM is shown to underperform SVM_T since $P(y)$ of unlabeled differs significantly from that of labeled data at PCR = 0.1 and 0.9. Apart

from the extreme ends of PCR settings, it can also be observed that DA methods including SVM_ST and FA underperform SVM_T on some of the imbalance PCR settings, indicating the presence of negative transfer. This is because both SVM_ST and FA require predictive distribution of the source and target domains to be similar, but the predictive distribution of imbalance PCR settings is quite diverse. From Table 3, when the predictive distribution of the source domains is similar to target domain (i.e. PCR≈0.5), SVM_S, SVM_ST and FA are shown to outperform SVM_T on all datasets. This implies that additional source labeled data can be useful for improving testing accuracy when the predictive distribution between source and target domain matches.

In all PCR settings, SVM_B is reported with better accuracies than many DA methods including SVM_ST, FA, M-SVM and DASVM. This implies re-sampling of the source domains to match the target predictive distribution is important for transfer learning to work well. In contrast, PDM-SVM can be observed to outperform all other classifiers, implying the predictive distribution matching of source and target domains in the PDM-SVM is deemed to be effective. In particular, even under extreme conditions of PCR settings in the source domain, PDM-SVM reported up to 28% accuracy improvements over the other classifiers.

As shown in Figure 3, each classifier displayed similar performance trends on the subgraph where most of the classifiers (excluding LapSVM, DASVM and PDM-SVM) showed sharp declining accuracies when the PCR is skewed toward either extremes. It can be observed that SVM_ST, FA and SVM_S gave the best accuracy of around 75% at PCR=0.5 and the worst accuracy in the

Table 3. Testing accuracies on Sentiment data set for PCR at 0.3, 0.5 and 0.7 in the source domains. The values below the accuracy results are the standard deviation.

Target	PCR	SVM_T	SVM_S	SVM_B	TSVM	LapSVM	SVM_ST	M-SVM	FA	DASVM	PDM-SVM
Book	0.3	58.51	59.91	69.11	68.65	68.89	62.4	57.8	56.27	66.25	**74.47**
		±2.31	±1.25	±1.34	±0.71	±1.24	±1.37	±3.12	±1.48	±4.61	±2.06
	0.5	58.51	70.88	71.77	72.8	71.14	71.77	69.08	71.48	65.35	**74.23**
		±2.31	±1.57	±1.24	±1.19	±1.28	±1.24	±1.55	±1.73	±2.93	±1.17
	0.7	58.51	58.98	69.11	69.39	61.15	61.05	58.96	55.9	62.4	**72.71**
		±2.31	±1.18	±1.34	±0.93	±2.57	±1.27	±2.23	±1.11	±1.11	±1.58
DVDs	0.3	60.1	60.73	72.1	70.05	70.74	63.7	59.74	56.74	67.86	**75.64**
		±2.40	±2.16	±1.24	±0.94	±1.21	±2.24	±3.02	±1.84	±3.61	±1.32
	0.5	60.1	73	73.24	74.3	73.06	73.24	68.75	73.18	71.58	**75.94**
		±2.40	±1.07	±0.93	±1.34	±1.18	±0.93	±2.05	±0.90	±3.51	±1.46
	0.7	60.1	61.04	72.1	71.38	63.03	63.45	60.9	57.53	67.01	**75.19**
		±2.40	±1.86	±1.24	±0.88	±2.55	±2.01	±1.76	±1.44	±6.94	±3.32
Electronic	0.3	61.78	61.54	74.37	73.35	74.42	65.64	61.7	60.07	71.55	**78.55**
		±2.56	±2.34	±1.10	±1.10	±0.93	±1.88	±1.69	±1.95	±2.48	±0.98
	0.5	61.78	74.67	75.36	78.79	75.08	75.36	70.03	75.17	74.88	**78.84**
		±2.56	±1.85	±1.67	±1.14	±1.48	±1.67	±2.59	±1.63	±1.65	±1.05
	0.7	61.78	68.52	74.37	73.7	65.22	71.11	63.67	65.47	72.01	**77.76**
		±2.56	±1.70	±1.10	±1.17	±1.29	±1.64	±2.26	±2.68	±3.11	±1.03
Kitchen	0.3	62.47	67.16	75.8	75.29	74.72	70.84	63.49	64.23	69.91	**79.06**
		±2.62	±1.83	±1.38	±0.75	±1.32	±1.75	±2.59	±1.91	±2.01	±1.12
	0.5	62.47	77.96	78.34	80.94	77.88	78.34	74.85	78.29	75.91	**81.15**
		±2.62	±1.01	±0.96	±1.28	±0.86	±0.96	±0.98	±1.16	±2.06	±1.34
	0.7	62.47	65.88	75.8	74.39	64.75	68.97	62.26	62.7	74.72	**80.01**
		±2.62	±1.96	±1.38	±0.91	±2.37	±1.50	±1.37	±2.27	±3.81	±1.68

region of 50% at PCR=0.1 and 0.9. Note that this marks a large difference in accuracies of up to 25%. Other methods also displayed significant variance in accuracy under the diverse PCR settings. As opposed to existing approaches suffering in performances due to the effect of negative transfer, on the other hand, PDM-SVM can effectively identify useful knowledge from multi-source domains by means of prediction distribution matching, thus achieving robust prediction performances in the target domain on the Sentiment data. In particular, PDM-SVM can still give readily stable results, and the accuracies are within 5% across all the PCR settings. This demonstrates the robustness of PDM-SVM under different PCR settings by benefiting from the positive transfer.

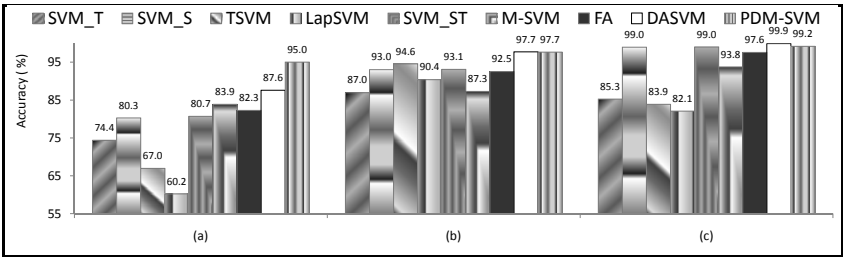

Fig. 4. Testing accuracies on Reuters data sets. (a)*People* versus *Organizations* (b)*People* versus *Exchanges* (c)*Organizations* versus *Exchanges*

Last but not least, we further experiment the classifiers on Reuters dataset with uneven class distribution in each domain. The results are reported in Figure 4. It can be observed that both PDM-SVM and DASVM had outperformed all other classifiers considered, see Figure 4(b,c). PDM-SVM on the other hand is competitive to DASVM. Interestingly, Figure 4(a) also indicated that PDM-SVM attained significant improvement in accuracy over DASVM. In all the experiments, SVM_S is shown to be competitive to some DA methods: SVM_ST, M-SVM and FA. It appears that most of the labels in the source domains are consistent with the target domain. This may be the reason why DASVM had performed well on the Reuters dataset while most DA methods outperforming SSL methods and SVM_T, whereas LapSVM and TSVM outperformed the other counterparts on Sentiment dataset. SSL methods on the other hand had performed much worse than the others on Reuters dataset. This is likely due to the manifold assumption and cluster assumption failing to hold on Reuters dataset. Overall, PDM-SVM is able to perform robustly and outperform all classifiers considered on both datasets, due to success of the predictive distribution matching regularizer in the identification of relevant data from source domains.

5 Conclusion

In this paper, we have presented a formalization of predictive distribution matching for addressing the effects of differing predictive distributions between related

domains. We address this problem by leveraging multiple domains to identify high predictive density regions, in which the class label represents the target class label in the same regions. Furthermore, we also present how to estimate the predictive distribution $P^t(y|\mathbf{x})$ of the target domain by using progressive transduction. On the other hand, empirical results obtained showed that while most DA methods suffer from the effect of negative transfer when the problem domains have mismatched predictive distributions, the proposed PDM-SVM reported robust prediction accuracy for diverse levels of PCR results on the dataset considered. In addition, PDM-SVM is shown capable of generating a substantial improvement over existing methods in most cases.

Acknowledgement

This research was supported by Singapore NTU AcRF Tier-1 Research Grant (RG15/08) and NTU and IHPC joint project entitled "Large Scale Domain Adaptation Machines: Information Integration, Revolution and Transfer".

References

1. Belkin, M., Niyogi, P., Sindhwani, V.: Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. JMLR 12, 2399–2434 (2006)
2. Bickel, S., Brückner, M., Scheffer, T.: Discriminative learning under covariate shift. JMLR 10, 2137–2155 (2009)
3. Blitzer, J., Dredze, M., Pereira, F.: Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In: ACL (2007)
4. Blitzer, J., McDonald, R., Pereira, F.: Domain adaptation with structural correspondence learning. In: EMNLP (2006)
5. Bruzzone, L., Marconcini, M.: Domain adaptation problems: A dasvm classification technique and a circular validation strategy. IEEE Trans. on PAMI 32(5), 770–787 (2010)
6. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: Synthetic minority over-sampling technique. JAIR 16, 321–357 (2002)
7. Chen, Y., Wang, G., Dong, S.: Learning with progressive transductive support vector machine. In: ICDM (2002)
8. Crammer, K., Kearns, M., Wortman, J.: Learning from multiple sources. JMLR 9, 1757–1774 (2008)
9. Daumé III., H.: Frustratingly easy domain adaptation. In: ACL (2007)
10. Duan, L., Tsang, I.W., Xu, D., Chua, T.S.: Domain adaptation from multiple sources via auxiliary classifiers. In: ICML (2009)
11. Duan, L., Tsang, I.W., Xu, D., Maybank, S.J.: Domain transfer svm for video concept detection. In: CVPR (2009)
12. Huang, J., Smola, A., Gretton, A., Borgwardt, K.M., Schölkopf, B.: Correcting sample selection bias by unlabeled data. In: NIPS (2006)
13. Jiang, J., Zhai, C.: Instance weighting for domain adaptation in NLP. In: ACL (2007)

14. Jiang, J., Zhai, C.: A two-stage approach to domain adaptation for statistical classifiers. In: CIKM (2007)
15. Joachims, T.: Transductive inference for text classification using support vector machines. In: ICML (1999)
16. Luo, P., Zhuang, F., Xiong, H., Xiong, Y., He, Q.: Transfer learning from multiple source domains via consensus regularization. In: CIKM (2008)
17. Pan, S.J., Tsang, I.W., Kwok, J.T., Yang, Q.: Domain adaptation via transfer component analysis. In: IJCAI (2009)
18. Pan, S.J., Ni, X., Sun, J.T., Yang, Q., Chen, Z.: Cross-domain sentiment classification via spectral feature alignment. In: WWW (2010)
19. Rosenstein, M.T., Marx, Z., Kaelbling, L.P.: To transfer or not to transfer. In: NIPS 2005 Workshop on Inductive Transfer: 10 Years Later (2005)
20. Schweikert, G., Widmer, C., Schölkopf, B., Rätsch, G.: An empirical analysis of domain adaptation algorithm for genomic sequence analysis. In: NIPS (2009)
21. Wu, P., Dietterich, T.G.: Improving SVM accuracy by training on auxiliary data sources. In: ICML (2004)
22. Zhu, X.: Semi-supervised learning literature survey. Technical report, Computer Sciences Technique Report 1530, University of Wisconsin-Madison (2009)