2011 11th IEEE International Conference on Data Mining

Healing Sample Selection Bias by Source Classifier Selection

Chun-Wei Seah, Ivor Wai-Hung Tsang, Yew-Soon Ong
School of Computer Engineering, Nanyang Technological University, Singapore, 639798
Email: {seah0116,IvorTsang,ASYSOng}@ntu.edu.sg

***Abstract*—Domain Adaptation (DA) methods are usually carried out by means of simply reducing the *marginal distribution* differences between the source and target domains, and subsequently using the resultant trained classifier, namely source classifier, for use in the target domain. However, in many cases, the true *predictive distributions* of the source and target domains can be vastly different especially when their class distributions are skewed, causing the issues of *sample selection bias* in DA. Hence, DA methods which leverage the source labeled data may suffer from poor generalization in the target domain, resulting in *negative transfer*. In addition, we observed that many DA methods use either a source classifier or a linear combination of source classifiers with a fixed weighting for predicting the target unlabeled data. Essentially, the labels of the target unlabeled data are spanned by the prediction of these source classifiers. Motivated by these observations, in this paper, we propose to construct many source classifiers of diverse biases and learn the weight for each source classifier by directly minimizing the structural risk defined on the target unlabeled data so as to heal the possible sample selection bias. Since the weights are learned by maximizing the margin of separation between opposite classes on the target unlabeled data, the proposed method is established here as *Maximal Margin Target Label Learning (MMTLL)*, which is in a form of Multiple Kernel Learning problem with many label kernels. Extensive experimental studies of MMTLL against several state-of-the-art methods on the *Sentiment* and *Newsgroups* datasets with various imbalanced class settings showed that MMTLL exhibited robust accuracies on all the settings considered and was resilient to negative transfer, in contrast to other counterpart methods which suffered significantly in prediction accuracy.**

***Keywords*-Domain Adaptation, Sample Selection Bias, Negative Transfer, Maximum Margin Separation, Multiple Kernel Learning, Classifier Selection**

I. Introduction

To date, many practical realizations of machine intelligence and classification are making their way as important tools that assist humans in their decision making process. A motivating example is sentiment prediction which assists marketing personnel in their formulations of novel sale strategies. In many instances, when a new product is launched, many comments may be posted over the Internet without providing any sentiment polarity, i.e., positive or negative feedbacks. Without any label information, typical supervised and semi-supervised techniques [1] are unable to be applied directly. To address this problem, *Domain Adaptation* (DA) learning methods have been introduced to leverage labeled samples from *source domains* [2]. Here, *target domain* refers to the current task to be solved, while *source domains* refer to the tasks that bear certain similarities to the target domain.

Ideally, the target model θ is learned from a predefined hypothesis space $\mathcal{H}$ by minimizing the following expected risk functional [2]:

$$\min_{\theta\in\mathcal{H}} \int L(\mathbf{x}, y, \theta)\mathrm{d}P^T(\mathbf{x}, y), \tag{1}$$

where L is the loss function and $P^T(\mathbf{x}, y)$ is the joint distribution of the input $\mathbf{x} \in \mathcal{X}$ and output $y \in \{\pm 1\}$ in the target domain which, however, is not accessible in practice. Since $P^T(\mathbf{x}, y) = P^T(\mathbf{x})P^T(y|\mathbf{x})$ where $P^T(\mathbf{x})$ and $P^T(y|\mathbf{x})$ are the marginal and predictive distribution of the target domain, respectively. By using the source predictive distribution, $P^S(y|\mathbf{x})$, to approximate the target predictive distribution, *i.e.*, $P^S(y|\mathbf{x}) \approxeq P^T(y|\mathbf{x})$, DA methods attempt to learn the target model θ by minimizing the following empirical risk functional defined on the source labeled samples:

$$\min_{\theta\in\mathcal{H}} \sum_{i=1}^{n_s} L(\mathbf{x}_i, y_i, \theta)P^S(\mathbf{x}_i)r(\mathbf{x}_i)P^S(y_i|\mathbf{x}_i), \tag{2}$$

where $r(\mathbf{x}) = \frac{P^T(\mathbf{x})}{P^S(\mathbf{x})}$, which is the ratio of the target marginal distribution $P^T(\mathbf{x})$ to the source marginal distribution $P^S(\mathbf{x})$, can be estimated using the target unlabeled samples and source samples subject to some criteria [3]–[6], and n_s is the number of source labeled samples $\{\mathbf{x}_i, y_i\}^{n_s}$.

In general, the distribution of the source samples (training set) may considerably differ from that of the target samples (test set), and so their corresponding predictive distributions are also different, *i.e.*, $P^S(y|\mathbf{x}) \neq P^T(y|\mathbf{x})$, leading to *sample selection bias* in DA. This phenomenon further deteriorates when their class distributions differ from each other [7], [8]. As a result, such sample selection bias usually creeps in inevitably, which poses a negative impact on the generalization performance of SVM classifier and other popular classifiers in the target domain [9]. In such a scenario, as shown in [7], [8], most DA methods as well as semi-supervised approaches including TSVM [10] and LapSVM [11], which leverage the source labeled data in their learning process, also suffer from poor generalization in the target domain, which is often known as *negative transfer* [12].

To avoid sample selection bias in DA, one should minimize the expected risk functional (1) directly by employing

1550-4786/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDM.2011.73

some prior knowledge on the output label structure of the target domain. An intuitive solution is to group u target unlabeled samples by means of unsupervised learning subject to some criteria. For instance, Maximum Margin Clustering (MMC) [13], which maximizes the margin of separation between opposite clusters via any possible combinations of labeling on unlabeled samples. Hence, MMC is able to choose the labels of the unlabeled samples from c^u unique label combinations for a c class problem. However, the large number of label combinations may lead to a trivial solution such as grouping all the samples as positive which is deemed useless [13]–[15]. In addition, since MMC does not use any label information from the source domains, which can be harnessed to guide the method to good solution from the identified set of promising solutions. Hence, MMC may give inferior performance compared to DA methods.

On the other hand, we observed that many DA methods use either a classifier learned from single source domain [4], [5], [16] or a linear combination of classifiers learned from multiple source domains [17], [18] to infer the true labels of the target unlabeled data. Thus, in general, the hypothesis space of the labels of the target unlabeled data is essentially spanned by the outputs predicted by the classifiers learned from the source domains, referred to here as source classifiers.

Based on this observation, in this paper, we proposed to directly learn the labels of the target unlabeled data, which form a *label vector*. This label vector is assumed to be a linear combination of the outputs on the target unlabeled data predicted by some source classifiers (see Sec. III-A). Meanwhile, to alleviate the sample selection bias from the source labeled data, we learn the weight of this linear combination by minimizing the structural risk functional (similar to (1)) defined on the target unlabeled samples only. Since the outputs on the target unlabeled data predicted by existing source classifiers are given in advance, which in turn form a *label matrix* for the target unlabeled data. Hence, the resultant optimization problem can be reduced to a Multiple Kernel Learning (MKL) [19] problem with a set of given label matrices (see Sec. III-B), in which the optimal weight for each label matrix, as well as the final decision classifier can be learned simultaneously. In this paper, we employ the maximum margin criterion used in SVM [1] to group the target unlabeled data into discrete classes. Hence, our method is called *Maximal Margin Target Label Learning* (MMTLL).

The core contributions of the current paper are summarized and outlined as follows:

1) To our knowledge, this is the first DA work that learns the labels of the target unlabeled data from a convex hull of the outputs predicted by multiple source classifiers, namely label vectors. Unlike existing DA methods, which either use one source classifier or a linear combination of multiple source classifiers with a predefined weighting for prediction [17], [18], [20], our method can simultaneously learn the final target classifier and the weight for each source classifier such that some poor label vectors (with a small margin) will be pruned out and promising label vectors (with a large margin) will be re-weighted higher.
2) Existing DA methods usually train classifiers using (2) based on the source labeled samples so the issues pertaining to sample selection bias in DA remain to persist. In contrast, our method directly minimize the risk functional in (1) defined on the target unlabeled data only by using an appropriate combination of available source classifiers of different biases in a manner such that the sample selection bias issue caused by imbalanced class distribution is minimized. In addition, since only precomputed source classifiers are needed, MMTLL can also cater for the situation where only precomputed classifiers are provided, and the source labeled data are in private or are required to be preserved in their domain only. In the same spirit, our approach is also beneficial to the field involving large scale decentralized database, where only the precomputed classifiers need to be moved and learned instead of the huge datasets during target predictions.
3) In the experimental study, we showed that MMTLL emerged as superior to several state-of-the-art DA methods in most of the tasks considered and was resilient to negative transfer [12] whereas other methods suffered for some of the settings.

The rest of this paper is organized as follows: Section II discusses the related work. Section III introduces MMTLL and its implementation. Extensive experiments on Sentiment and Newsgroup datasets are carried out in Section IV. Then the experimental results are analyzed and discussed in Section V. Lastly, the conclusive remarks of this paper are drawn in Section VI.

II. Related Work

In [2], the authors showed that DA learning problems can be solved by minimizing the *empirical risk* of the target domain in (1) if the joint distributions of the target domain is known. However, in practice, labeled data are usually very limited or even absent in the target domain, hence existing DA methods generally estimate the joint distribution of the target domain from the source domains.

The initial work of DA using a single source domain was proposed in [21] by assuming that the joint distribution of the source domain is the same as that of the target domain. An extension of the work to multiple source domains was subsequently proposed [17]. The approach is established as *Multiple Convex Combinations* (MCC), which formulates each source domain as a Support Vector Machines (SVM)

[1]Other criteria such as maximum likelihood, maximum entropy can also be used for grouping the target unlabeled data.

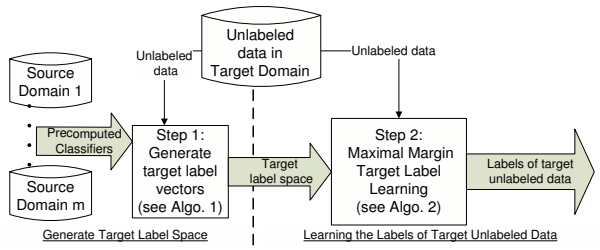


Figure 1. Maximal Margin Target Label Learning Framework

learning problem while treating all the source domains equally. However, directly applying all the source domains to the learning task can be harmful for predicting the target data [22]. Instead of treating all classifiers equally, a further extension of MCC, namely *Domain Adaptation Machine* (DAM) [18], was proposed to incorporate some prior knowledge between the source and target domains in order to define the importance of each source classifier.

In [4], *Kernel-Mean Matching* (KMM) was proposed to estimate the marginal distribution of the target domain by learning the weight of each source sample; then a classifier is trained by minimizing the empirical risk of the re-weighted source samples in (2). In spite of the advancements made, KMM nevertheless assumes the predictive distributions, $p(y|\mathbf{x})$, between the source and target domains to be similar. Taking this cue, in this paper, we present a study that relaxes the degree of similarity in the predictive distributions of the source and target domains.

Another popular trend to alleviate the effect of sample selection bias in DA is to find an appropriate feature representation of the source domains that would represent the feature space of the target domain well [23], [24]. Most recently, *Transfer Component Analysis* (TCA) [25] is proposed to identify a suitable latent space spanned by some basis vectors, referred to as *transfer components*. Since the aforementioned methods usually train a classifier or learn a model θ by minimizing the empirical risk of the source domain in (2), the issues pertaining to sample selection bias remain to persist.

Most recent alternatives to address samples selection bias are Domain Adaptation SVM (DASVM) [7], Bridging Information Gap (BIG) [26] and Predictive Distribution Matching SVM (PDMSVM) [8]. Such approaches involve expensive transductive learning processes to estimate the joint distribution of the target domain by assigning *pseudo-labels* to a set of target unlabeled samples iteratively. The main differences between these methods and our proposed MMTLL are listed as follows. These methods start with learning a large margin classifier from some source domains. After that, the current classifier is used to choose some confident target unlabeled data for assigning pseudo-labels and adding them into the training set. Then a new classifier is trained. The whole process is repeated until some stopping criteria are fulfilled. Whereas MMTLL directly finds the target classifier from a target label space, which is spanned by a convex hull of the source classifiers with different biases, and at the same time to maximize the margin of separation on the target unlabeled data only. In addition, the learning process of MMTLL does not involve any computationally expensive transductive process and is able to obtain the globally optimal solution.

Algorithm 1 Generating a set of label vectors for the target label space

Inputs: F (a set of precomputed classifiers trained from each unique combination of source domains)
Outputs: Y (a set of generated label vectors)
$s = 1$;
for all $f \in F$ **do**
  indexes=sort($f(\mathbf{x}_i), ..., f(\mathbf{x}_u)$) //impose the balance constraint by sorting
  for $q = \beta$ to $u - \beta$ **do**
    create a column vector $\mathbf{y}_s \in \Re^u$
    assign $\mathbf{y}_s$ with the first q *indexes* as negative (-1) and the rest as positive $(+1)$
    $Y = Y \cup \mathbf{y}_s$; $s = s + 1$;
  end for
end for
return Y

III. Maximal Margin Target Label Learning

Throughout the rest of this paper, superscript $'$ denotes the transpose of a vector or matrix and $\mathbf{1}$ defines a vector with all ones. Given m source domains and one target domain $\mathcal{X}_u$, which contains u unlabeled(testing) samples, $\mathbf{x}_j$'s, the task of Domain Adaptation (DA) is to predict the class label, $\hat{y}_j$, for each unlabeled sample in the target domain by leveraging labeled data in the source domains.

Figure 1 depicts the learning process of MMTLL framework. First, from each combination of source domains, a precomputed classifier can be trained using any DA methods. Nevertheless, the bias of the precomputed classifier learned from the source domains would lead to sample selection bias. To address this bias issue, MMTLL learns the bias of each precomputed classifier by using the target unlabeled data, meanwhile MMTLL also uses these precomputed classifiers to generate numerous source classifiers to form the target label space (see Sec. III-A). With the target label space, MMTLL maximizes the margin of separation of the target unlabeled data in the label space that is spanned by a linear combination of source classifiers and then proceeds to learn the weight of each source classifier (see Sec. III-B).

In what follows, we will discuss the relations between MMTLL and existing DA methods: MCC, DAM and CP-MDA. In single source domain setting, DA methods learn a classifier, f, and predict a target unlabeled sample $\mathbf{x}_i \in \mathcal{X}_u$ as $\hat{y}_i = \text{sign}(f(\mathbf{x}_i))$ where $\text{sign}(\cdot)$ is the sign function such that $\text{sign}(t) = 1$ if $t > 0$; otherwise, $\text{sign}(t) = -1$. In [17], MCC is proposed for multiple source domains

and predicts $\mathbf{x}_i$ as $\hat{y}_i = \text{sign}\big(\frac{1}{m}\sum_{s=1}^{m} f_s(\mathbf{x}_i)\big)$ where f_s is the classifier trained on the sth source domain. And each f_s is treated equally. In [18], the weight g_s of each classifier is predefined based on the distribution differences between the source and target domains. Then, DAM predicts $\mathbf{x}_i$ using $\hat{y}_i = \text{sign}\big(\sum_{s=1}^{m} g_s f_s(\mathbf{x}_i)\big)$ with $\sum_{s=1}^{m} g_s = 1$. Finally, a target classifier is learned based on $\hat{y}_i$. Chattopadhyay *et al.* [20] proposed Conditional Probability based Multi-source Domain Adaptation (CP-MDA), which extends DAM with an additional manifold regularizer defined on the target unlabeled data. First, the weight g_s of each classifier is precomputed, then a target classifier is learned based on these weights. In contrast, the task of MMTLL is to directly learn the weight g_s and the target learning model simultaneously. Hence, the solution of many DA methods can be deemed as a special case of the MMTLL framework.

A. Generating Target Label Space via Multiple Source Classifiers

For m source domains, a precomputed classifier can be trained for each individual source domain and every combination of $2, 3, ..., (m-1)$ source domains, till all m source domains are combined; thus $\sum_{i=1}^{m} \frac{m!}{i!(m-i)!}$ classifiers can be trained. Normally, each precomputed classifier includes a bias b so that the decision boundary is not restricted to intersect at the origin. Since these classifiers are trained from the source domains which bear differing distributions to the target domain, it would generally be more appropriate to determine the bias based on target domain $\mathcal{X}_u$. Herein, we propose to learn the bias from the target unlabeled samples by imposing a balance constraint such that $\beta \leq \mathbf{1}'(\mathbf{y}_s + \mathbf{1})/2 \leq u - \beta$, where $\mathbf{y}_s = [\hat{y}_1, \cdots, \hat{y}_u]'$ and β is the parameter to control the class balance. Then this balance constraint can be implemented by sorting the precomputed classifier's decision outputs on the target unlabeled data. And in turn there are $u - 2\beta$ ways to impose the balance constraint, resulting in $(u-2\beta)\sum_{i=1}^{m} \frac{m!}{i!(m-i)!}$ label vectors. Notice that diverse forms of precomputed classifiers can be trained using SVM, Gaussian Process, Transductive SVM [10] or other supervised, semi-supervised and DA methods. For simplicity, we only consider supervised SVM method in the present study. The overall algorithm is outlined in Algorithm 1.

Upon Z label vectors are formed, the target label space, $\mathcal{M}$, is defined as follows:

$$\begin{aligned}\mathcal{M} \quad = \quad & \big\{\hat{\mathbf{y}} = \textstyle\sum_{s=1}^{Z} g_s \mathbf{y}_s \big| \sum_{s=1}^{Z} g_s = 1,\ g_s \geq 0, \\ & \beta \leq \mathbf{1}'(\mathbf{y}_s + \mathbf{1})/2 \leq (u-\beta),\ \forall s = 1, ..., Z\big\}\end{aligned} \tag{3}$$

which forms a convex hull [27] of the label vectors of the target unlabeled data, where the importance of each label vector, $\mathbf{y}_s$, is weighted by g_s. Note, the balance constraint, $\beta \leq \mathbf{1}'(\mathbf{y}_s + \mathbf{1})/2 \leq u - \beta$, is enforced by Algorithm 1 while generating $\mathbf{y}_s$.

B. Learning the Labels of Target Unlabeled Data

Since the true distributions of the source and target domains often differ, negative transfer may result. Taking this cue, instead of minimizing (2), we proposed to minimize the expected risk functional in (1) using only the target unlabeled samples with the loss function L [2]. In particular, MMTLL maximizes the margin of separation with the regularizer $\|\mathbf{w}\|_2^2$; meanwhile at the same time, MMTLL learns the labels of the target unlabeled samples to minimize the structural risk functional:

$$\begin{aligned}\min_{\hat{\mathbf{y}} \in \mathcal{M}} \quad & \Big\{ \min_{\mathbf{w}, \rho, \boldsymbol{\xi}} \quad \frac{1}{2}\|\mathbf{w}\|_2^2 - \rho + C\sum_{i=1}^{u} \xi_i \\ & \qquad s.t. \qquad \hat{y}_i \mathbf{w}'\phi(\mathbf{x}_i) \geq \rho - \xi_i, \forall i = 1, ..., u \Big\}\end{aligned} \tag{4}$$

where $\phi(\cdot)$ maps $\mathbf{x}_i$ into a high dimensional space induced by a kernel k, the decision function is denoted as $\mathbf{w}'\phi(\mathbf{x})$, $\rho/\|\mathbf{w}\|$ is the margin of separation and C denotes the regularization parameter that controls the model complexity ($\|\mathbf{w}\|$) and the empirical risk (the slack variables ξ_i's).

MMTLL learns the weight of each label vector (from the source classifier) $\mathbf{y}_s$ in (4) by minimizing the structural risk of the samples in the target domain only. This is in contrast to DAM [18] and its variant, CP-MDA [20], which require prior knowledge about the source and target domains to determine the weight of each label vector $\mathbf{y}_s$. Furthermore, learning the target classifier in DAM or CP-MDA requires some target labeled data. It is notable that here, MMTLL does not impose such requirement on the target labeled data. Nevertheless, if some target labeled data are available, MMTLL can easily incorporate them by fixing the labels of the target labeled data in each $\mathbf{y}_s$.

In what follows, the steps to solve (4) will be described. First, the dual of the inner minimization in (4) is as follows:

$$\min_{\hat{\mathbf{y}} \in \mathcal{M}} \quad \Big\{ \max_{\boldsymbol{\alpha} \in \mathcal{A}} \quad -\frac{1}{2}\boldsymbol{\alpha}'(\mathbf{K} \odot \hat{\mathbf{y}}\hat{\mathbf{y}}')\boldsymbol{\alpha} \Big\} \tag{5}$$

where $\boldsymbol{\alpha}$ is the vector of the Lagrangian multipliers for the inequalities in (4), $\mathcal{A} = \{\boldsymbol{\alpha} | \sum_{i=1}^{u} \alpha_i = 1, 0 \leq \alpha_i \leq C, \forall i = 1, ..., u\}$, $\mathbf{K} = [k(\mathbf{x}_i, \mathbf{x}_j)]_{u \times u}$ is the kernel matrix where $k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)'\phi(\mathbf{x}_j)$, and $\odot$ denotes the element-wise product operator. Since $\mathcal{A}$ and $\mathcal{M}$ are both compact sets and according to minimax theorem [28], swapping the order of the $\min$ and $\max$ in (5) is equivalent to:

$$\max_{\boldsymbol{\alpha} \in \mathcal{A}} \quad \Big\{ \min_{\hat{\mathbf{y}} \in \mathcal{M}} \quad -\frac{1}{2}\boldsymbol{\alpha}'(\mathbf{K} \odot \hat{\mathbf{y}}\hat{\mathbf{y}}')\boldsymbol{\alpha} \Big\} \tag{6}$$

In addition, (6) can be reformulated as:

$$\begin{aligned}\max_{\boldsymbol{\alpha} \in \mathcal{A}} \quad & \Big\{ \max_{\theta} \quad -\theta \\ & \qquad s.t. \qquad \theta \geq \frac{1}{2}\boldsymbol{\alpha}'(\mathbf{K} \odot \mathbf{y}_t {\mathbf{y}_t}')\boldsymbol{\alpha}, \forall \mathbf{y}_t \in \mathcal{M} \Big\}\end{aligned} \tag{7}$$

[2]For simplicity, in this paper, we use the hinge loss function in SVM, but we can also use logistic loss or square loss in the proposed framework.

Moreover, the dual form of the inner maximization of (7) is:

$$\max_{\boldsymbol{\alpha}\in\mathcal{A}} \Big\{ \min_{\boldsymbol{d}\in\mathcal{D}} \; -\frac{1}{2}\boldsymbol{\alpha}'\Big(\sum_{t:\mathbf{y}_t\in M} d_t \mathbf{K}\odot \mathbf{y}_t\mathbf{y}_t{}'\Big)\boldsymbol{\alpha}\Big\} \qquad (8)$$

where $\mathbf{d}$ is the vector of the Lagrangian multipliers d_t's for the inequalities in (7) and $\mathcal{D} = \{\boldsymbol{d} | \sum_{t:\mathbf{y}_t\in\mathcal{M}} d_t = 1, d_t \geq 0,\ \forall t : \mathbf{y}_t \in \mathcal{M}\}$. Since $\mathcal{D}$ and $\mathcal{A}$ are compact, swapping the order of the $\max$ and $\min$ in (8) is equivalent to:

$$\min_{\boldsymbol{d}\in\mathcal{D}} \Big\{ \max_{\boldsymbol{\alpha}\in\mathcal{A}} \; -\frac{1}{2}\boldsymbol{\alpha}'\Big(\sum_{t:\mathbf{y}_t\in M} d_t \mathbf{K}\odot \mathbf{y}_t\mathbf{y}_t{}'\Big)\boldsymbol{\alpha}\Big\} \qquad (9)$$

Note, (9) can be deemed as a Multiple Kernel Learning (MKL) problem [19] where each of the base kernels in MKL is represented by $\mathbf{K}\odot\mathbf{y}_t\mathbf{y}_t^{'}$. Hence, (9) can be solved using any efficient MKL solver where $\min_{\boldsymbol{d}\in\mathcal{D}}$ and $\max_{\boldsymbol{\alpha}\in A}$ can be solved iteratively.

Due to the presence of a large number of source classifiers, solving (9) via MKL may not be efficient. Fortunately, since it is unlikely for all of the constraints in (7) to be active at the optimal solution, the efficient cutting plane method can be used [15], [29], [30] to solve (9) (see Algorithm 2). The algorithm begins with the initialization of $\boldsymbol{\alpha} = \frac{1}{u}\mathbf{1}$ and then locates the most violated constraint which is the one with largest objective value in (10) and fails the constraint in (7).

Theorem 1. *The most violated constraint of* (7) *for a fixed* $\boldsymbol{\alpha}$ *is then:*

$$\arg\max_{\mathbf{y}\in\mathcal{M}_2} \quad \frac{1}{2}\mathbf{y}'(\mathbf{K}\odot\boldsymbol{\alpha}\boldsymbol{\alpha}')\mathbf{y}\ ,\ where\ \mathcal{M}_2 = \{\mathbf{y}_1, ..., \mathbf{y}_Z\} \qquad (10)$$

Proof: Let $F(\mathbf{y}) = \frac{1}{2}\mathbf{y}'(\mathbf{K}\odot\boldsymbol{\alpha}\boldsymbol{\alpha}')\mathbf{y}$. Since $F()$ is a convex function, $F((1-\lambda)\mathbf{y}_i + \lambda\mathbf{y}_j) \leq (1-\lambda)F(\mathbf{y}_i) + \lambda F(\mathbf{y}_j), \forall \mathbf{y}_i, \mathbf{y}_j \in \mathcal{M}_2, \lambda \in [0,1]$ according to the convexity property. Suppose $F(\mathbf{y}_i) > F(\mathbf{y}_j)$, then we have $F((1-\lambda)\mathbf{y}_i + \lambda\mathbf{y}_j) \leq F(\mathbf{y}_i)$. Similarity, if $F(\mathbf{y}_i) < F(\mathbf{y}_j)$, then $F((1-\lambda)\mathbf{y}_i + \lambda\mathbf{y}_j) \leq F(\mathbf{y}_j)$. Therefore, $F((1-\lambda)\mathbf{y}_i + \lambda\mathbf{y}_j) \leq \max(F(\mathbf{y}_i), F(\mathbf{y}_j))$ holds. Using induction [27], $F(\lambda_1\mathbf{y}_1 + \lambda_2\mathbf{y}_2 + ... + \lambda_Z\mathbf{y}_Z) \leq (\arg\max_{\mathbf{y}\in\mathcal{M}_2} f(\mathbf{y}))$ given $\sum_{i=1}^{Z}\lambda_i = 1$ and $\forall\lambda_i \in [0,1]$. ■

Note that for (10), no numerical optimization solver is needed as the maximum objective value is simply obtained by computing all the objective values in the set $\mathcal{M}_2$. And the most violated $\mathbf{y}_t$ corresponds to the one with the highest value among those computed. Hence, the active constraint is chosen based on the most violated $\mathbf{y}_t$. Then, the current set of selected constraints are solved via MKL. The process of finding the next most violated constraint is repeated until convergence. Empirically, only a few iterations is needed for Algorithm 2 to converge. Assuming the empirical complexity of the SVM training is $O(u^{2.3})$, the overall time complexity of MMTLL is $O(TJ(u^{2.3}))$, where J and T are iterations incurred by the cutting plane method and MKL, respectively. From our experiments, J is generally less than a dozen and T is usually small as it depends on J.

Algorithm 2 Maximal Margin Target Label Learning (MMTLL)

Inputs: $\mathcal{M}_2$ //a set of label vectors
$\boldsymbol{\alpha} = \frac{1}{u}\mathbf{1}$, then find the most violated $\mathbf{y}_t$ in (10) and let $\mathcal{S} = \{\mathbf{y}_t\}$
repeat
  Find the optimal $\boldsymbol{d} \in \mathcal{S}$ and $\boldsymbol{\alpha}$ in (9) via MKL
  Find the most violated $\mathbf{y}_t$ by (10) and set $\mathcal{S} = \mathcal{S} \cup \mathbf{y}_t$
until convergence
return $Y = \sum_{t:\mathbf{y}_t\in\mathcal{S}} d_t\mathbf{y}_t\mathbf{y}_t{}'$

Table I
Grouping of source and target domains in Newsgroup dataset

Domain	Category comp	Category rec	Category sci
Source 1	windows.x	motorcycles	electronics
Source 2	sys.ibm.pc.hardware	sport.baseball	med
Source 3	sys.mac.hardware	sport.hockey	space
Target	graphics	autos	crypt

Upon convergence, we have $\sum_{t:\mathbf{y}_t\in M} d_t \odot \mathbf{y}_t\mathbf{y}_t{}' = \mathbf{Y}\text{diag}(\mathbf{d})\mathbf{Y}' = \mathbf{Y}\text{diag}(\mathbf{d})^{0.5}(\mathbf{Y}\text{diag}(\mathbf{d})^{0.5})'$, where $\mathbf{Y} = [\mathbf{y}_1, \cdots, \mathbf{y}_Z]$ and $\text{diag}(\mathbf{d})$ returns a diagonal matrix with $\mathbf{d}$ as the diagonal entries. Following [15], the labels of the target unlabeled data can be recovered using the eigenvector $\boldsymbol{V_1}$ corresponding to the largest singular value D_1 of $\mathbf{Y}\text{diag}(\mathbf{d})^{0.5}$ by means of singular value decomposition, which takes $O(uZ^2)$ time complexity. The polarity of the groups can then be determined by the majority vote of the precomputed classifiers.

IV. Experimental Study

In the present study, several state-of-the-art algorithms are investigated on different settings with the datasets involving three source domains and a target domain:

1) *1S-SVM$_{Best}$*: Each source domain is trained using SVM and the best accuracy among the classifiers is reported.
2) *2S-SVM$_{Best}$*: Each unique pair of source domains is trained using SVM and the best accuracy among the classifiers is reported.
3) *MCC*: Multiple Convex Combination denotes a representative of DA method that linearly combines all source domains and trained using SVM [17]. Since the present study involves three source domains, MCC is equivalent to 3S-SVM.
4) *LG-MMC*: Label Generating Maximum Margin Clustering[3] [15] maximizes the margin separating two opposite clusters of the target unlabeled data without using any label information of the source domains. Since LG-MMC does not use any class label information, we assume the class labels assigning to the respective clusters are the class labels that will give the best accuracy.

[3]The program is downloaded from http://lamda.nju.edu.cn/files/LGMMC_v2.rar

5) *KMM*$_{Best}$: Kernel Mean Matching addresses the marginal distribution differences between a single source domain and a target domain by re-weighting each of the source samples in the Reproducing Kernel Hilbert Space (RKHS) such that the Maximum Mean Discrepancy (MMD) criterion defined with the source and target domains [4] is minimized. A weighted SVM is then trained on the source domain using the derived weight of each sample. One KMM is trained for each source domain and the best accuracy among the classifiers is reported.
6) *TCA*$_{Best}$: Transfer Component Analysis assumes there exists some feature map with similar predictive distributions between a single source domain and a target domain, i.e., $P^S(y|\mathbf{x}) \approx P^T(y|\mathbf{x})$, where superscripts ^{S} and ^{T} refer to source domain and target domain, respectively. Hence, it learns a set of transfer components in a RKHS using the MMD criterion, and then SVM is trained on the source domain in this RKHS [25]. One TCA is trained for each source domain and the best accuracy among the classifiers is reported.
7) *MMTLL*: Maximal Margin Target Label Learning learns the labels of the target unlabeled data through maximizing the margin separation of the target data based on the label space spanned by a linear combination of source classifiers described in Figure 1.

The parameters of all methods are configured by means of k-fold cross-source domains validation as suggested in [21] (an extension of k-fold cross validation for DA). Here, k is the number of source domains, i.e. $k = m$. Specifically, each partition represents a source domain in k-fold cross-source domains validation. In addition, β is fixed as $0.3 \times u$ in LG-MMC and MMTLL. In the experimental study, the datasets are pre-processed, with only the single-terms extracted, stopwords removed, stemming and normalizing of each feature performed. Consequently, each feature of the sample is represented by its respective *tf-idf* value and the linear kernel is employed.

As discussed in [7], [8], the class imbalance in the source and target domains would cause their joint distributions to differ further and this may lead to negative transfer. That is, the learned source classifiers have degraded performances in the target domains. In practice, since the true class distribution of the target domain is usually unknown, to analyze the effects of class imbalance in the source and target domains towards different learning algorithms, the term *Target Positive Class Ratio* (TPCR) is defined to denote the number of positive samples in the target domain. For example in a set of 1000 target samples, a TPCR of 0.3 implies 300 samples are positive and the rest are negative. In the experimental study, TPCR values of 0.3, 0.5 and 0.7 are investigated. Similarly, the term *Source Positive Class Ratio* (SPCR) is also defined to denote the number of positive samples in the source domain. In the experimental study, the robustness of different state-of-the-art algorithms for different configurations, particularly SPCR values of 0.2, 0.4, 0.6 and 0.8 are also investigated.

For imbalanced target class settings, Area under the Curve (AUC) performance measure is commonly used [31]. In addition, AUC defined by one run, i.e., a particular TPCR value, is also well known as balanced accuracy [32]. Furthermore, existing work that seeks to factor out the effect of class distribution also used balanced accuracy in their evaluation [33]. Hence, balanced accuracy is reported as the evaluation measure in the experimental study:

$$Balanced\ Accuracy = 0.5(\frac{tp}{tp+fn} + \frac{tn}{tn+fp}) \qquad (11)$$

where tp, fn, tn, fp are the number of true positive, false negative, true negative and false positive, respectively.

A. Multi-Domain Sentiment Dataset

The dataset was prepared in [24]. It comprises four categories of product reviews: *Book*, *DVDs*, *Electronics*, and *Kitchen appliances* from Amazon.com. Each review is marked with a five-star rating where a higher star rating implies a better feedback. The 3-star ratings data are removed to avoid ambiguity in binary classification, and the negative samples are made up of 1-star and 2-star ratings whereas the rest of the ratings form the positive samples. For each task, one category is posed as the target domain while the rest as related source domains. In each of the tasks, 2000 samples are randomly selected from each source domain to form the labeled data and 500 samples from the target domain as unlabeled data. The average results of 10 independent runs per task are then reported.

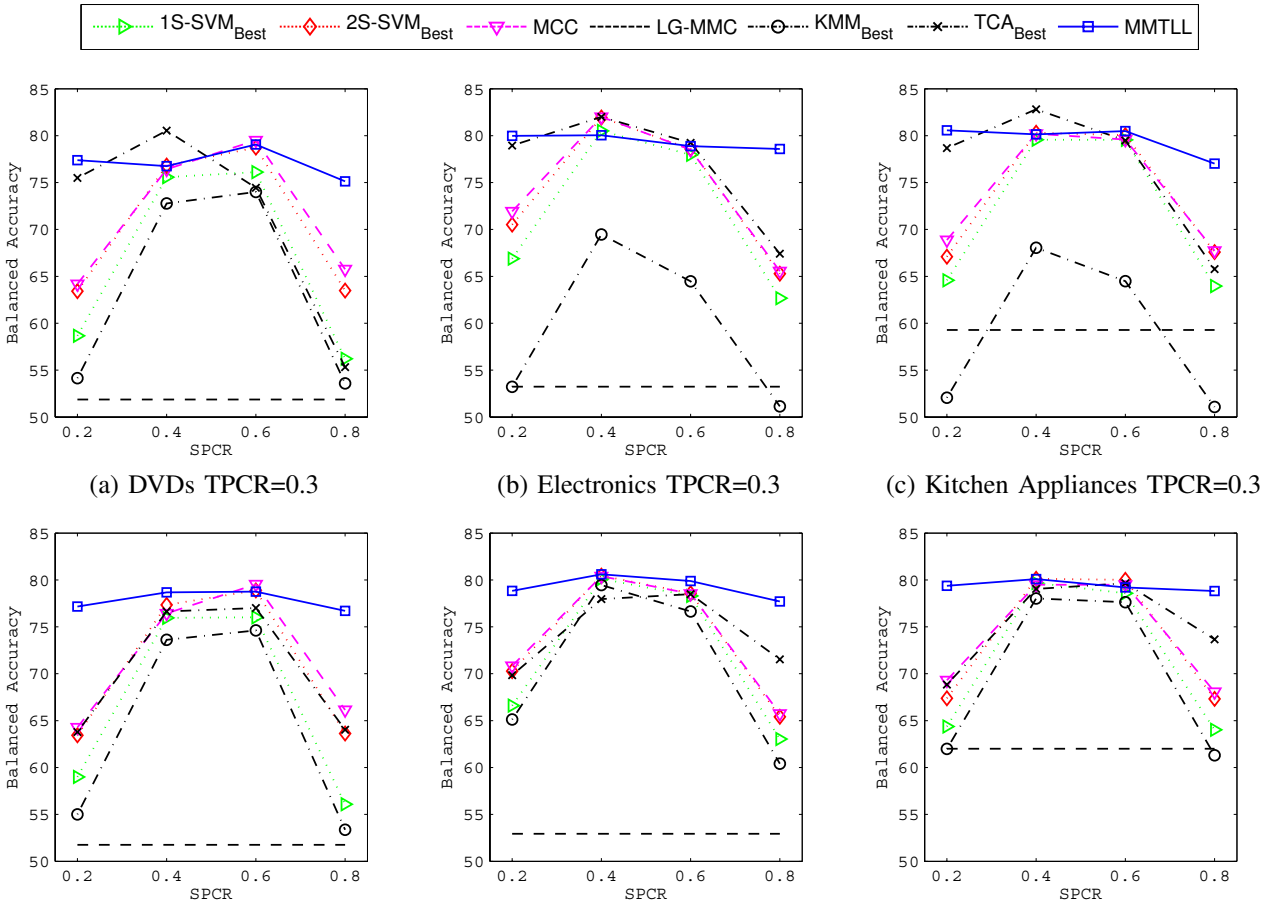
B. Multi-Domain Newsgroup Dataset

The dataset consists of three main categories: *comp*, *rec*, and *sci*. Each main category is separated into Source 1, Source 2, Source 3 and Target (see Table I), resulting in three tasks: *comp vs. rec*, *comp vs. sci* and *rec vs. sci*. In each of the tasks, 1000 samples are randomly selected from each source domain to form the labeled data while 500 samples from the target domain as unlabeled data. Similarly, each task is repeated 10 times and the average results are reported.

V. Results and Discussions

A. Sentiment Experimental Result Discussion

In Figure 2, the reported results for the Sentiment prediction dataset are the balanced accuracy defined in (11) on the target unlabeled data. The three subfigures on the top denote the results obtained on the target domain with a positive class ratio (TPCR) of 0.3 whereas the other three subfigures on the bottom represent the results for TPCR of 0.5. Subfigures 2(a) and 2(d) summarize the balanced accuracy of the target domain with varying SPCR in the source domain on the *DVDs* datasets. On the other hand, subfigures 2(b,e) and



(a) DVDs TPCR=0.3 (b) Electronics TPCR=0.3 (c) Kitchen Appliances TPCR=0.3

(d) DVDs TPCR=0.5 (e) Electronics TPCR=0.5 (f) Kitchen Appliances TPCR=0.5

Figure 2. Sentiment Experimental Results where top section having target domain's positive class ratio(TPCR) as 0.3 and the bottom section is TPCR=0.5. The x-axis is the various source domain's positive class ratio(SPCR) settings and the y-axis is the balanced accuracy. Maximal Margin Target Label Learning (MMTLL) is our proposed method.

subfigures 2(c,f) show the results for *Electronics* and *Kitchen* appliances datasets as the target domain, respectively. Note that due to space constraints, experimental results on the *Book* dataset as the target domain are omitted from this paper. Since the results for TPCR=0.7 is symmetrical to that of TPCR=0.3, all other target domains for TPCR of 0.7 are also omitted.

As observed from Figure 2, LG-MMC exhibited the worst balanced accuracy across all methods for most of the results reported. This indicates that an unsupervised approach based on maximal margin separation of the unlabeled data without using any label information is less effective than other DA methods that leverages the abundant labeled data from other related source domains for classifying the target unlabeled data. Thus, DA methods are useful on the Sentiment data in the absence of label information in the target domain.

When the SPCR approaches to the two extremes (*i.e.*, 0.2 and 0.8), the performances of most methods, except for MMTLL, significantly degraded. Since both KMM and TCA minimize the marginal distribution differences between the target and source domains using the MMD criterion [4], the degraded performances indicate that the necessary assumption on similar predictive distributions between source and target domains in KMM and TCA does not always hold on the Sentiment data. And this verifies that the change in class distribution would easily lead to sample selection bias.

With the TPCR varying between 0.3 to 0.5, the results show that the trends of most methods are unaffected by the changing of TPCR except for KMM_{Best} and TCA_{Best}. The former re-weights the importance of each sample and the latter finds a suitable kernel mapping between the source and target domains. In particular, on average, KMM_{Best} degrades more than 10% in balanced accuracy in all SPCR settings when TPCR varies from 0.5 to 0.3. On the other hand, by leveraging the source labeled data, TCA_{Best} reported improved balanced accuracies over 1S-SVM_{Best} for SPCR of 0.2 and 0.4 when TPCA changes from 0.5 to 0.3. This implies that TCA (feature based DA method) is more effective than KMM (instance weighting based DA method) on the Sentiment data due to harnessing the label information from the source domain despite the class distribution being skewed.

Recall that negative transfer is observed when a DA method makes use of source data but achieves degraded performances than other methods that do not use any source data. From Figure 2(c), the results of KMM$_{Best}$ for SPCR of 0.2 or 0.8 are noted to be significantly poorer than that of LG-MMC (unsupervised learning method). Thus, negative transfer is notable in KMM$_{Best}$ on the sentiment data.

The performances of other DA methods are also shown to suffer due to the sample selection biases of the source domains. The proposed MMTLL method, which maximizes the margin of separation only on the target unlabeled data via the label space that is spanned by the source classifiers with different biases, on the other hand is observed to perform robustly across the varying SPCR and TPCR settings. In addition, MMTLL is noted to have attained higher prediction accuracies than all the other methods for SPCR of 0.2 and 0.8, as observed in all the subfigures. This demonstrates the success of MMTLL in minimizing the sample selection bias on the Sentiment data across all settings considered.

It is worth noting that TCA$_{Best}$ also attains superior accuracy at SPCR of 0.4 on the Sentiment data (see Figure 2(a,b,c)). Note that the reported balanced accuracy of TCA$_{Best}$ is the best accuracy among the three results reported in Figure 3, each of which is obtained by applying TCA on different source domain. The prediction accuracy of each source domain trained using SVM, which is denoted as 1S-SVM is also depicted within Figure 3. In practice, it is non-trivial to determine which source domain is the most suitable for the target domain beforehand, especially in the absence of prior knowledge on the target domain. MMTLL thus fills this gap by suitable source classifiers and ensembles them for improved predictive performance in the target domain of interest. Therefore, in general, Figure 3 shows that TCA performs much worse than MMTLL throughout all SPCR settings.

B. Newsgroup Experimental Result Discussion

The results for the Newsgroup data are reported in Figure 5. We can observe that LG-MMC achieves decent performances on the Newsgroup data. Particularly, LG-MMC reports improved balanced accuracies over 1S-SVM$_{Best}$, 2S-SVM$_{Best}$, MCC and KMM$_{Best}$ for SPCR of 0.2 and 0.8 in most of the subfigures. This implies that learning from target unlabeled data only can be more beneficial than the labeled samples of other source domains, when the target data have well separated cluster structures. 1S-SVM$_{Best}$ also operates based on maximizing the margin of separation but training is concentrated on the source domain. It is observed to achieve higher prediction accuracies over LG-MMC at SPCR 0.4 and 0.6 on the *comp vs. rec* task. However, lower accuracies are reported relative to LG-MMC at SPCR 0.2 and 0.8. On the other hand, KMM$_{Best}$ improved the accuracy performance by matching the marginal distributions between the source and target domains. However, KMM$_{Best}$ still fares lower than LG-MMC. In addition, negative transfer is notable in all methods except MMTLL (see Figure 5(b,c,e,f)) since their prediction accuracies deteriorate over LG-MMC. Nevertheless, TCA$_{Best}$ and MMTLL significantly achieve higher balanced accuracies than LG-MMC, 1S-SVM$_{Best}$ and KMM$_{Best}$. Overall, MMTLL emerges as superior to all other methods considered in all experimental settings except on the *rec vs. sci* task where TPCR=0.3.

The results in Figure 4 indicates that TCA performs poorer relative to MMTLL if source 2 or source 3 is considered in the training process of classifying the target unlabeled data. Therefore, selecting which source domain for TCA is an essential task that will impact on its effectiveness. However, in practice, it is difficult to determine the most appropriate source domain for TCA beforehand.

C. Comparison with other Sample Selection Bias methods

In what follows, the focus is placed on DA methods that consider a transductive paradigm to address sample selection bias. Here the representative approaches considered are Domain Adaptation SVM (DASVM) [7] and Predictive Distribution Matching SVM (PDMSVM) [8]. The time complexity of DASVM and PDMSVM is $O(u(n_s)^{2.3})$ and $O(u(n_s+u)^3)$, respectively, where n_s is the total number of source samples in all domains and $O((n_s)^{2.3})$ denotes the empirical complexity assumed in the SVM training. In this paper, we are particularly interested on the multiple source domain setting which was demonstrated to be suitable for DA problems [8], [17], [18], [20]. Under this setting, usually $n_s \gg u$. Recall, the time complexity of MMTLL is $O(TJ(u^{2.3}))$. When $n_s \gg u$, both DASVM and PDMSVM are more computationally expensive than MMTLL. For the sake of conciseness, we only report the experimental results for DASVM and MMTLL on the Kitchen appliances, where the target domain has n_s values of 1000, 2000, 3000, 4000, 5000 and 6000 for TPCR of 0.3 and $u = 500$. Note, n_s source samples has balance labeled samples from each source domain. Similarly, each task is repeated 10 times and the average results are reported.

Due to space constraint, only the average time of SPCR of 0.8 is reported. Nevertheless, the average time for the rest of SPCR values are similar to SPCR of 0.8. In Figure 6, the time incurred by DASVM is noted to be 50 times more than MMTLL (including the training time for different source classifiers) at $n_s = 6000$. Hence, DASVM does not scale better than MMTLL when $n_s \gg u$. As n_s reduces gradually, the time incurred by DASVM also reduced. However, the balanced accuracy of DASVM also degrades sharply along with the decreasing n_s considered as shown in Table II. Furthermore, MMTLL is reported as superior to DASVM in all settings. In addition, the balanced accuracy of MMTLL remains robust for varying n_s. This is due to the success of MMTLL, which learns the labels from a convex hull of numerous source classifiers of different bias to identify

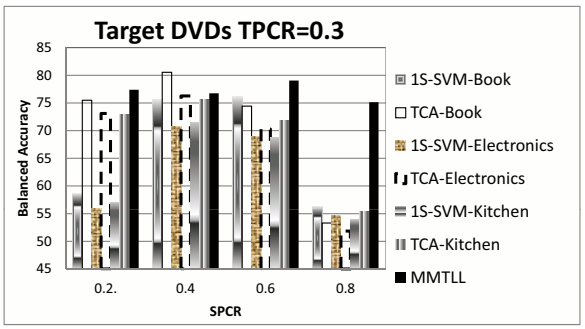


Figure 3. DVDs as Target Domain in Sentiment Experiments. Comparisons among 1S-SVM, TCA and MMTLL. 1S-SVM-*X* or TCA-*X* where *X* refers to the source domain being used to classify DVDs' test data.

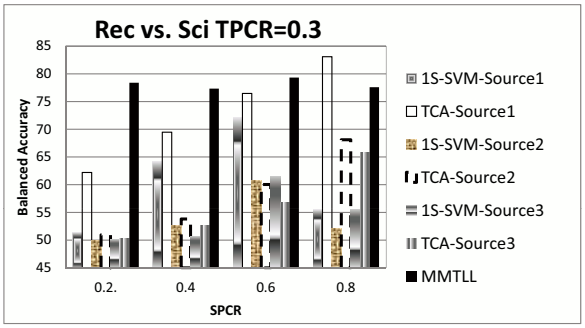


Figure 4. Rec vs. Sci in Newsgroup Experiments. Comparisons among 1S-SVM, TCA and MMTLL. 1S-SVM-*X* or TCA-*X* where *X* refers to the source domain (see Table I) being used to classify the target test data.

suitable source classifiers with distributions that are close to the target distribution for improved inference.

VI. Conclusion

Existing DA methods train classifier using samples in the source domain as part of their training set. Due to the differing distributions of the source and target domains, the issue of sample selection bias often creeps in, resulting in poor prediction accuracy of current DA methods. To address this, we propose the novel Maximal Margin Target Label Learning (MMTLL) method, which learns the important weight of each source classifier via maximizing the margin separation of the samples only in the target domain. In the experimental study, MMTLL is shown to display superiority across the entire range of imbalanced class settings when pit against several state-of-the-art methods. In addition, MMTLL is also resilient to negative transfer, in contrast to other counterpart methods which suffered significantly in prediction accuracy. Last but not least, MMTLL also exhibited robust accuracies on all the settings considered.

Acknowledgement

This research was in part supported by Singapore NTU AcRF Tier-1 Research Grant (RG15/08) and A* SERC Grant (102 158 0034).

References

[1] B. Schölkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond (Adaptive Computation and Machine Learning)*, 2001.

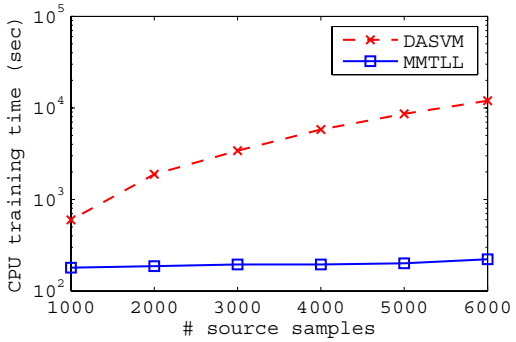
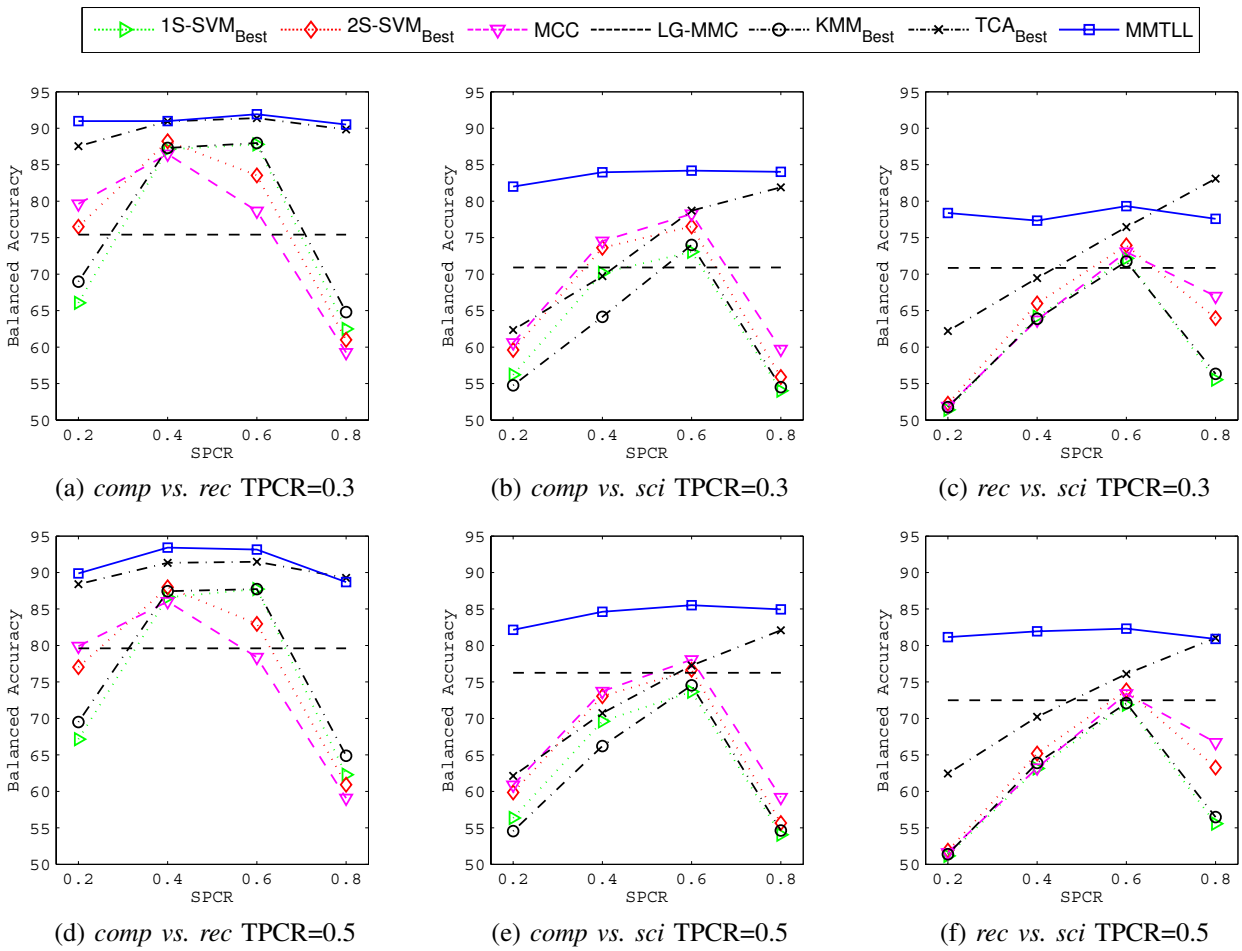


Figure 6. Time performance on Kitchen appliances as target domain at TPCR=0.3 and SPCR=0.8

Table II
Balanced Accuracy on Kitchen appliances as target domain at TPCR=0.3

SPCR	DA method	source dataset size 6000	5000	4000	3000	2000	1000
0.2	MMTLL	**80.57**	**79.00**	**79.26**	**78.78**	**77.37**	**74.40**
	DASVM	75.02	73.96	74.25	71.73	66.71	56.55
0.4	MMTLL	**80.15**	**80.13**	**80.78**	**79.28**	**78.92**	**77.06**
	DASVM	79.99	79.44	79.22	77.77	76.62	66.10
0.6	MMTLL	**80.49**	**79.63**	**79.55**	**77.12**	**78.06**	**76.39**
	DASVM	78.58	78.06	77.18	76.49	75.56	71.22
0.8	MMTLL	**77.07**	**78.10**	**78.07**	**78.70**	**74.93**	**72.61**
	DASVM	74.02	73.87	72.68	71.85	67.68	60.98

[2] S. J. Pan and Q. Yang, "A Survey on Transfer Learning," *IEEE TKDE*, vol. 22, pp. 1345–1359, 2010.
[3] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira, "Analysis of representations for domain adaptation," in *NIPS*, 2006.
[4] J. Huang, A. Smola, A. Gretton, K. M. Borgwardt, and B. Schölkopf, "Correcting Sample Selection Bias by Unlabeled Data," in *NIPS*, 2006.
[5] M. Sugiyama, S. Nakajima, H. Kashima, P. von Bunau, and M. Kawanabe, "Direct Importance Estimation with Model Selection and Its Application to Covariate Shift Adaptation," in *NIPS*, 2007.
[6] B. Chen, W. Lam, I. W. Tsang, and T.-L. Wong, "Location and Scatter Matching for Dataset Shift in Text Mining," in *ICDM*, 2010.
[7] L. Bruzzone and M. Marconcini, "Domain Adaptation Problems: A DASVM Classification Technique and a Circular Validation Strategy," *IEEE Trans. on PAMI*, vol. 32, no. 5, pp. 770–787, 2010.
[8] C.-W. Seah, I. W. Tsang, Y.-S. Ong, and K.-K. Lee, "Predictive Distribution Matching SVM for Multi-domain Learning," in *ECML/PKDD*, 2010.
[9] B. Z. Zadrozny, "Learning and Evaluating Classifiers under Sample Selection Bias," in *ICML*, 2004, pp. 903–910.
[10] T. Joachims, "Transductive Inference for Text Classification using Support Vector Machines," in *ICML*, 1999.
[11] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold Regularization: A Geometric Framework for Learning from Labeled and Unlabeled Examples," *JMLR*, vol. 12, pp. 2399–2434, 2006.
[12] M. T. Rosenstein, Z. Marx, and L. P. Kaelbling, "To transfer or not to transfer," in *NIPS 2005 Workshop on Inductive Transfer: 10 Years Later*, 2005.
[13] L. Xu, J. Neufeld, B. Larson, and D. Schuurmans, "Maximum margin clustering," in *NIPS*, 2005, pp. 1537–1544.
[14] K. Zhang, I. W. Tsang, and J. T. Kwok, "Maximum margin clustering made practical," in *ICML*, 2007, pp. 1119–1126.
[15] Y.-F. Li, I. W. Tsang, J. T. Kwok, and Z.-H. Zhou, "Tighter and convex maximum margin clustering," in *AISTATS*, 2009, pp. 328–335.



(a) *comp vs. rec* TPCR=0.3 (b) *comp vs. sci* TPCR=0.3 (c) *rec vs. sci* TPCR=0.3

(d) *comp vs. rec* TPCR=0.5 (e) *comp vs. sci* TPCR=0.5 (f) *rec vs. sci* TPCR=0.5

Figure 5. Newsgroup Experimental Results where top section having target domain's positive class ratio(TPCR) as 0.3 and the bottom section is TPCR=0.5. The x-axis is the various source domain's positive class ratio(SPCR) settings and the y-axis is the balanced accuracy. Maximal Margin Target Label Learning (MMTLL) is our proposed method.

[16] S. Bickel, C. Sawade, and T. Scheffer, "Transfer Learning by Distribution Matching for Targeted Advertising," in *NIPS*, 2008, pp. 145–152.

[17] G. Schweikert, C. Widmer, B. Schölkopf, and G. Rätsch, "An Empirical Analysis of Domain Adaptation Algorithm for Genomic Sequence Analysis," in *NIPS*, 2009.

[18] L. Duan, I. W. Tsang, D. Xu, and T. S. Chua, "Domain Adaptation from Multiple Sources via Auxiliary Classifiers," in *ICML*, 2009.

[19] G. Lanckriet, N. Cristianini, P. Bartlett, and L. E. Ghaoui, "Learning the kernel matrix with semidefinite programming," *JMLR*, pp. 27–72, 2004.

[20] R. Chattopadhyay, J. Ye, S. Panchanathan, W. Fan, "Multi-Source Domain Adaptation and Its Application to Early Detection of Fatigue," in *KDD*, 2011.

[21] P. Wu and T. G. Dietterich, "Improving SVM Accuracy by Training on Auxiliary Data Sources," in *ICML*, 2004.

[22] X. Shi, Q. Liu, W. Fan, Q. Yang, and P. S. Yu, "Predictive Modeling with Heterogeneous Sources," in *ICDM*, 2010, pp. 814–825.

[23] J. Blitzer, R. McDonald, and F. Pereira, "Domain Adaptation with Structural Correspondence Learning," in *EMNLP*, 2006.

[24] J. Blitzer, M. Dredze, and F. Pereira, "Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification," in *ACL*, 2007.

[25] S. J. Pan, I. Tsang, J. Kwok, and Q. Yang, "Domain Adaptation via Transfer Component Analysis," *IEEE TNN*, vol. 22, pp. 199 – 210, 2011.

[26] E. W. Xiang, B. Cao, D. H. Hu, and Q. Yang, "Bridging Domains Using World Wide Knowledge for Transfer Learning," *IEEE TKDE*, pp. 770–783, 2010.

[27] S. Boyd and L. Vandenberghe, *Convex Optimization*, 2004.

[28] S.-J. Kim and S. Boyd, "A Minimax Theorem with Applications to Machine Learning, Signal Processing, and Finance," *SIAM J. on Optimization*, vol. 19, pp. 1344–1367, 2008.

[29] J. Kelley, J. E., "The Cutting-Plane Method for Solving Convex Programs," *SIAM*, vol. 8, pp. 703–712, 1960.

[30] M. Tan, L. Wang, and I. W. Tsang, "Learning Sparse SVM for Feature Selection on Very High Dimensional Datasets," in *ICML*, 2010, pp. 1047–1054.

[31] N. V. Chawla, N. Japkowicz, and A. Kotcz, "Editorial: special issue on learning from imbalanced data sets," *SIGKDD Explor. Newsl.*, vol. 6, pp. 1–6, 2004.

[32] M. Sokolova, N. Japkowicz, and S. Szpakowicz, "Beyond Accuracy, F-Score and ROC: A Family of Discriminant Measures for Performance Evaluation," *AI*, pp. 1015–1021, 2006.

[33] G. Forman, "Counting Positives Accurately Despite Inaccurate Classification," in *ECML/PKDD*, 2005, pp. 564–575.