

Multi-View Transfer Learning with a Large Margin Approach

Dan Zhang
Computer Science
Department
Purdue University
West Lafayette, IN
zhang168@cs.purdue.edu

Jingrui He
Machine Learning Group
IBM TJ Watson Research
Center
Yorktown Heights, NY
jingruhe@us.ibm.com

Yan Liu
Computer Science
Department
University of Southern
California
Los Angeles, CA
yanliu.cs@usc.edu

Luo Si
Computer Science
Department
Purdue University
West Lafayette, IN
lsi@cs.purdue.edu

Richard D. Lawrence
Machine Learning Group
IBM TJ Watson Research
Center
Yorktown Heights, NY
ricklawr@us.ibm.com

ABSTRACT

Transfer learning has been proposed to address the problem of scarcity of labeled data in the target domain by leveraging the data from the source domain. In many real world applications, data is often represented from different perspectives, which correspond to multiple views. For example, a web page can be described by its contents and its associated links. However, most existing transfer learning methods fail to capture the multi-view nature, and might not be best suited for such applications.

To better leverage both the labeled data from the source domain and the features from different views, this paper proposes a general framework: Multi-View Transfer Learning with a Large Margin Approach (MVTL-LM). On one hand, labeled data from the source domain is effectively utilized to construct a large margin classifier; on the other hand, the data from both domains is employed to impose consistencies among multiple views. As an instantiation of this framework, we propose an efficient optimization method, which is guaranteed to converge to ϵ precision in $O(1/\epsilon)$ steps. Furthermore, we analyze its error bound, which improves over existing results of related methods. An extensive set of experiments are conducted to demonstrate the advantages of our proposed method over state-of-the-art techniques.

Categories and Subject Descriptors

I.2.6 [Artificial Intelligence]: Learning—*Knowledge acquisition*

General Terms

Algorithms, Performance, Experimentation

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

KDD'11, August 21–24, 2011, San Diego, California, USA.
Copyright 2011 ACM 978-1-4503-0813-7/11/08 ...\$10.00.

Keywords

Multi-View Learning, Transfer Learning, Large Margin Approach

1. INTRODUCTION

Transfer learning has been proposed and commonly used to address the problem of scarcity of labeled data in a particular target domain. It builds a model for the target domain by leveraging the label information from another related domain (source domain), thus avoids the costly process of generating labels for target domain examples. In many real world applications, the examples are often described from different perspectives, which correspond to multiple views. For example, in web mining, a web page can be represented by its contents and in-bound/out-bound links; in image analysis, an image can be described by different types of features, such as, color, texture, and shape. It has been shown that leveraging the consistencies between different views can help improve the learning performance. However, most existing transfer learning methods are designed only for single-view problems. In other words, if applied to problems with multiple views, these methods would fail to utilize the redundancy incurred by the multi-view property. Therefore, they are not ideal for such applications.

Despite its importance, the problem of Multi-View Transfer Learning (MVTL) has received limited attention. Most existing methods integrate the multi-view and transfer learning nature by some heuristics without theoretical analysis, such as the convergence rate of their algorithms, and how the learned model for the target domain can be improved by integrating the characteristics of multi-view features into transfer learning. And researchers tend to put more emphasis on the multi-view side. For example, in [40], the proposed algorithm uses the classifier trained on the source domain to generate the initial seed set, and then applies the co-training [4] algorithm to construct the classifier for the target domain; in [13], the authors explicitly model the view consistency in their objective function without considering the data distribution difference between the source domain and the target domain. Another straightforward method for MVTL is to concatenate the features from multiple views, and apply the transfer learning methods for single-view problems to build the model for the target domain. However, this kind of methods disregard the redundancy incurred by different

views. As shown in a lot of previous works [37, 44, 47], without considering the consistencies between different views, the performance of multi-view learning cannot be guaranteed. On the contrary, in MVTL, we utilize the nature of both the transfer learning and the multi-view learning in a unified way.

To achieve this goal, this paper proposes a general framework: Multi-View Transfer Learning with a Large Margin Approach (MVTL-LM). In particular, from the transfer learning perspective, we integrate the nature of the multi-view setting into the transfer learning framework and impose the consistencies among multiple views, which implicitly limits the capacity of the hypothesis class. This is significantly different from previous large margin transfer learning methods [29], which do not consider the problem of multi-view setting. From multi-view learning perspective, the new formulation explicitly models the data distribution difference between the source domain and the target domain, applies the technique of importance sampling [1], and uses weighted labeled data from the source domain to construct a large margin classifier for the target domain. In particular, we propose a novel optimization method based on the bundle method [38, 39], which can solve an instantiation of the MVTL-LM framework in a very efficient way. We further prove that the optimization method can converge to ϵ precision in $O(\frac{1}{\epsilon})$ steps (See Theorem 3.1) and each step takes time $O(sn)$, where s is the average sparsity for features and n is the total number of source domain examples (See Theorem 3.2). Moreover, we analyze the generalized error bound of MVTL-LM in the target domain, which depends on its performance in the source domain and the empirical Rademacher complexity of a related hypothesis class (See Theorem 3.3). The empirical Rademacher complexity is reduced by making use of the multi-view nature. Experimental results demonstrate the advantages of our proposed method over state-of-the-art techniques.

The rest of this paper is organized as follows. The related works are discussed in Section 2. Section 3 introduces the research problem, presents the proposed algorithm and analyzes some properties of the proposed method. Section 4 presents the experimental results. Section 5 concludes the whole paper.

2. RELATED WORKS

This section briefly introduces related works on transfer learning, multi-view learning, MVTL and the bundle method.

2.1 Transfer Learning

As an important technique to address the problem of scarcity of labeled data in the target domain, transfer learning has received much attention recently. A key problem in transfer learning is what kind of knowledge can be transferred from the source domain to the target domain. Roughly speaking, the assumptions introduced in previous transfer learning work can be grouped into four categories:

1. **Feature Representation Transfer.** In [2, 7, 9, 11, 28, 30], the authors assume that there exists a common feature space shared by both the source domain and the target domain, which can be used as a bridge to transfer knowledge.
2. **Parameter Transfer.** In [5, 19], the authors make use of Gaussian Process (GP) models, and assume that the source domain and the target domain have shared parameters / hyper-parameters.
3. **Instance Transfer.** Due to the data distribution difference between the source domain and the target domain, in [8, 42], the authors select or re-weight the examples from the source domain for use in the target domain such that the expectation

of the adjusted loss function on the source domain examples can be as close to the expectation of the loss function on the target domain examples as possible. In [23, 24], the authors utilize the data distribution differences between several source domains and the target domain, and propose a method to combine the classifiers from these source domains optimally.

4. **Relation Transfer.** In [10, 25, 26], the authors build the relational map between the source and the target domains, and relax the independently and identically distributed (i.e., iid) assumption in these two domains.

Despite their success on single-view problems, existing transfer learning methods may not work well on MVTL problems since the nature of multiple views are not considered in these works. In contrast, besides transferring knowledge from the source domain to the target domain via instance transfer, our proposed MVTL-LM approach imposes the consistencies between multiple views by explicitly modeling the differences between the outputs from different views, which promises to improve the performance.

2.2 Multi-View Learning

In many real-world applications, examples are represented by multiple views. It has been shown extensively in prior research that leveraging the redundancy among the multiple views can improve the learning performance [12, 17, 20, 31, 36, 44, 45]. For example, the authors of [12] construct a classifier on each view and regulate the consistencies between different views. Furthermore, they show that the Rademacher complexity of the function class can also be greatly reduced by regulating the consistencies.

This idea is further exploited in [20], where the authors incorporate the consistency term into multi-view semi-supervised learning problems, and show a substantial improvement on the classification performance. Similarly, in [44], the authors incorporate this idea into local learning [41] and propose a novel way to define the graph Laplacian. Most existing multi-view learning methods are for the single-domain settings. However, in MVTL problems, the source domain and the target domain do not have the same data distribution. As we will show in the experiments, disregarding the data distribution difference may adversely affect the classification performance. In contrast, besides leveraging the consistency between different views, our proposed MVTL-LM approach also considers the domain difference by an effective re-weighting scheme, so it can achieve better performance in MVTL problems.

2.3 Multi-View Transfer Learning

As mentioned in introduction, existing methods for multi-view transfer learning combine the multi-view learning and transfer learning by some heuristics, and tend to put more emphasis on the multi-view side. For example, the co-adaptation algorithm proposed in [40] uses the labeled examples from the source domain to construct classifiers, which will be used to generate the initial seed set. It then applies the co-training algorithm [4] to construct the classifier for the target domain. Note that in the latter stage, labeled examples from the source domain are not utilized, which may otherwise provide useful insights about consistency between multiple views and the optimal classifier. In [13], the co-regularized loss function consists of the standard regularized log likelihood on multiple views based on the labeled data, as well as the expected Bhattacharyya distance based on the unlabeled data. When applied in MVTL problems, it may fail to capture the data distribution difference between the source domain and the target domain. In contrast, our proposed MVTL-LM approach integrates the multi-view

and transfer learning nature in a principled way, and is tailored for MVTL problems. Furthermore, we address some important theoretical problems, such as the convergence rate, time complexity and the generalization error bound, which have not been discussed in previous works for MVTL.

2.4 Bundle Method

The proposed formulation is a convex optimization problem. In this paper, we propose an optimization algorithm based on the bundle method [38, 39], which has shown its superior performances in both efficiency and effectiveness over state-of-the-art methods, to solve this proposed formulation. The basic motivation of the bundle method is to approximate the objective function $J(\mathbf{w})$ through a set of linear functions, where \mathbf{w} is the model parameter. In particular, this objective function is lower bounded as follows:

$$J(\mathbf{w}) \geq \max_{1 \leq i \leq t} \{J(\mathbf{w}_{i-1}) + \langle \mathbf{w} - \mathbf{w}_{i-1}, \mathbf{a}_i \rangle\},$$

where \mathbf{w}_i is a set of points picked by the bundle method, and \mathbf{a}_i is the gradient/sub-gradient at point \mathbf{w}_i . The bundle method monotonically decreases the gap between $J(\mathbf{w})$ and $\max_{1 \leq i \leq t} \{J(\mathbf{w}_{i-1}) + \langle \mathbf{w} - \mathbf{w}_{i-1}, \mathbf{a}_i \rangle\}$ such that the minimal point of $J(\mathbf{w})$ can be approximated by that of the line segments $\max_{1 \leq i \leq t} \{J(\mathbf{w}_{i-1}) + \langle \mathbf{w} - \mathbf{w}_{i-1}, \mathbf{a}_i \rangle\}$.

Some recent developments in bundle method [39] show that if $J(\mathbf{w})$ contains some regularizers by itself, the bundle method is guaranteed to converge to the precision ϵ in $O(1/\epsilon)$ steps. In MVTL-LM, we adapt the bundle method to solve the proposed problem, which can also be proven to have an efficient convergence rate.

3. MULTI-VIEW TRANSFER LEARNING WITH A LARGE MARGIN APPROACH

In this section, we first introduce the problem statement and some notations for MVTL. Then, a general framework named MVTL-LM is proposed, which integrates the multi-view and transfer learning nature in a principled way. Based on an instantiation of the framework, we propose an optimization method, which is adapted from the bundle method [38, 39]. Towards the end of this section, we analyze some important properties of the proposed method.

3.1 Problem Statement and Notations

Suppose we are given a set of labeled source domain examples from M independent views: $\{(\mathbf{x}_1^{(p)}, y_1^S), \dots, (\mathbf{x}_n^{(p)}, y_n^S)\}$, $\mathbf{x}_i^{(p)} \in \mathbf{R}^{d_p \times 1}$, $y_i^S \in \{-1, 1\}$, $p \in \{1, 2, \dots, M\}$, where n is the total number of source domain examples and d_p is the dimensionality of the p -th view. y_i^S is the class label of $\mathbf{x}_i^{(p)}$. Besides the source domain examples, a set of unlabeled target domain examples are also available, and are denoted as: $\{\mathbf{z}_1^{(p)}, \mathbf{z}_2^{(p)}, \dots, \mathbf{z}_m^{(p)}\}$, $\mathbf{z}_i^{(p)} \in \mathbf{R}^{d_p \times 1}$, $p \in \{1, 2, \dots, M\}$.

The goal of MVTL is to construct an accurate classifier for the target domain by making use of the labeled examples from the source domain as well as the redundancy incurred by multiple views. In this paper, similar to [15], we assume that $Pr_T(y|\mathbf{x}) = Pr_S(y|\mathbf{x})$. In other words, the conditional probability of the class label given the features is the same for both the source domain and the target domain. This particular case can also be referred to as covariate shift [35]. As claimed in [15], even in some cases when this assumption does not hold, the algorithm, which is based on this assumption, can still perform well.

3.2 MVTL-LM Framework

In this subsection, we propose a general large margin framework for MVTL, which fully exploits the multi-view and transfer learn-

ing nature. In this framework, we construct linear classifiers for all of the views, whose weight vectors are obtained via the following optimization problem:

$$\begin{aligned} \min_{\mathbf{w}^{(1)}, \dots, \mathbf{w}^{(M)}} & \sum_{p=1}^M \gamma_p \Omega(\mathbf{w}^{(p)}) + \sum_{p=1}^M C_p R(Pr_T, l(\mathbf{x}^{(p)}, y, \mathbf{w}^{(p)})) \\ & + \sum_{p=1}^M \sum_{q=1}^M C_{p,q} R_c(Pr_T^{(p,q)}, l_c(\mathbf{x}^{(p)}, \mathbf{x}^{(q)}, \mathbf{w}^{(p)}, \mathbf{w}^{(q)})), \end{aligned} \quad (1)$$

where $\Omega(\mathbf{w}^{(p)})$ is the regularization term defined on the p -th view weight vector $\mathbf{w}^{(p)}$; $R(Pr_T, l(\mathbf{x}^{(p)}, y, \mathbf{w}^{(p)}))$ is the expected classification loss with respect to the data distribution of the target domain examples (Pr_T), which measures the deviations between the true labels and the predicted labels based on the p -th view; $l(\mathbf{x}^{(p)}, y, \mathbf{w}^{(p)})$ is the classification loss (such as the hinge loss [32]); $R_c(Pr_T^{(p,q)}, l_c(\mathbf{x}^{(p)}, \mathbf{x}^{(q)}, \mathbf{w}^{(p)}, \mathbf{w}^{(q)}))$ measures the expected consistency between the p -th view and the q -th view with respect to their joint distribution in the target domain; $l_c(\mathbf{x}^{(p)}, \mathbf{x}^{(q)}, \mathbf{w}^{(p)}, \mathbf{w}^{(q)})$ is the consistency between the p -th view and the q -th view (such as the squared loss between the predictions on different views); γ_p , C_p , and $C_{p,q}$ are non-negative parameters that balance the relative importance of the three terms in the objective function in Eq.(1).

Notice that when $C_{p,q} = 0$, Eq.(1) is equivalent to training a large margin classifier on each view independently. When $C_{p,q} > 0$, by minimizing Eq.(1), we can obtain large margin classifiers which are also consistent across different views. The final classifier is the average of large margin classifiers on all the views, i.e.,

$$f(\mathbf{x}) = \frac{1}{M} \sum_{p=1}^M (\mathbf{w}^{(p)})^T \mathbf{x}^{(p)},$$

where $\mathbf{x} = [(\mathbf{x}^{(1)})^T, \dots, (\mathbf{x}^{(M)})^T]^T$. Next, we will discuss the loss term and the consistency term respectively.

3.2.1 Classification Loss

$R(Pr_T, l(\mathbf{x}^{(p)}, y, \mathbf{w}^{(p)}))$ measures the expected classification loss with respect to the data distribution on the target domain. To be specific, $R(Pr_T, l(\mathbf{x}^{(p)}, y, \mathbf{w}^{(p)})) = \mathbf{E}_{(\mathbf{x}, y) \sim Pr_T} [l(\mathbf{x}^{(p)}, y, \mathbf{w}^{(p)})]$. However, in our problem setting, we do not have any labeled examples from the target domain. Therefore, we estimate this term using labeled examples from the source domain as follows:

$$\begin{aligned} & \mathbf{E}_{(\mathbf{x}, y) \sim Pr_T} [l(\mathbf{x}^{(p)}, y, \mathbf{w}^{(p)})] \\ & = \mathbf{E}_{(\mathbf{x}, y) \sim Pr_S} \left[\frac{Pr_T(\mathbf{x}, y)}{Pr_S(\mathbf{x}, y)} l(\mathbf{x}^{(p)}, y, \mathbf{w}^{(p)}) \right] \\ & = \mathbf{E}_{(\mathbf{x}, y) \sim Pr_S} \left[\frac{Pr_T(y|\mathbf{x}) Pr_T(\mathbf{x})}{Pr_S(y|\mathbf{x}) Pr_S(\mathbf{x})} l(\mathbf{x}^{(p)}, y, \mathbf{w}^{(p)}) \right] \\ & = \mathbf{E}_{(\mathbf{x}, y) \sim Pr_S} \left[\frac{Pr_T(\mathbf{x})}{Pr_S(\mathbf{x})} l(\mathbf{x}^{(p)}, y, \mathbf{w}^{(p)}) \right] \\ & = \mathbf{E}_{(\mathbf{x}, y) \sim Pr_S} [\beta(\mathbf{x}) l(\mathbf{x}^{(p)}, y, \mathbf{w}^{(p)})] \\ & \approx \frac{1}{n} \sum_{i=1}^n \beta_i l(\mathbf{x}_i^{(p)}, y_i, \mathbf{w}^{(p)}), \end{aligned} \quad (2)$$

where Pr_S is the distribution of the source domain, weight function $\beta(\mathbf{x}) := \frac{Pr_T(\mathbf{x})}{Pr_S(\mathbf{x})}$; and $\beta_i = \beta(\mathbf{x}_i)$. Notice that the value of $\beta(\mathbf{x})$ reflects the distribution difference between the source domain and the target domain. If the two distributions are similar, $\beta(\mathbf{x})$ will be close to 1; if the two distributions are dissimilar, under-represented examples in Pr_T will receive a higher weight, whereas over-represented examples will receive a lower weight. In this way,

we are able to estimate the classification error for the target domain using labeled examples from the source domain.

There are various ways to estimate β_i , such as Gaussian Mixture Model (GMM) [22], Kernel Density Estimation [34], Kernel Mean Matching [15], etc. In our approach, we concatenate the features on different views together, and measure the probability ratios between source and target domains by using GMM. To be specific, we first estimate the marginal distribution of all the features in the source domain and the target domain using two GMMs respectively. Then, β_i is estimated as the ratio between the two generative probabilities on \mathbf{x}_i given by these two GMMs. Notice that in GMM, we need to specify the number of components. As will be shown in Section 4, the proposed approach is very robust to small perturbations in this number.

3.2.2 Consistency

As a consistency term, $R_c(P_{r_T}^{(p,q)}, l_c(\mathbf{x}^{(p)}, \mathbf{x}^{(q)}, \mathbf{w}^{(p)}, \mathbf{w}^{(q)}))$ regulates that the outputs on individual views should be consistent, and not deviate too much. $l_c(\mathbf{x}^{(p)}, \mathbf{x}^{(q)}, \mathbf{w}^{(p)}, \mathbf{w}^{(q)})$ is the consistency loss function, which penalizes the deviations between the output of $\mathbf{x}^{(p)}$ and $\mathbf{x}^{(q)}$, under the classifiers $\mathbf{w}^{(p)}$ and $\mathbf{w}^{(q)}$. Similar to Eq.(2), To estimate this term, we use both the labeled examples from the source domain and the unlabeled examples from the target domain as follows:

$$\begin{aligned} & R_c(P_{r_T}^{(p,q)}, l_c(\mathbf{x}^{(p)}, \mathbf{x}^{(q)}, \mathbf{w}^{(p)}, \mathbf{w}^{(q)})) \\ &= \mathbf{E}_{(\mathbf{x}^{(p)}, \mathbf{x}^{(q)}) \sim P_{r_T}^{(p,q)}} [l_c(\mathbf{x}^{(p)}, \mathbf{x}^{(q)}, \mathbf{w}^{(p)}, \mathbf{w}^{(q)})] \\ &\approx \frac{1}{m+n} \left(\sum_{i=1}^n \beta_i l_c(\mathbf{x}_i^{(p)}, \mathbf{x}_i^{(q)}, \mathbf{w}^{(p)}, \mathbf{w}^{(q)}) \right. \\ &\quad \left. + \sum_{i=1}^m l_c(\mathbf{z}_i^{(p)}, \mathbf{z}_i^{(q)}, \mathbf{w}^{(p)}, \mathbf{w}^{(q)}) \right). \end{aligned} \quad (3)$$

This term regulates the consistency on both the source domain and target domain examples. Combining this constraint with the standard objective functions for each view yields a multi-view learning algorithm, which was shown to perform better than single view approach on many classification applications.

3.3 Method

In this section, a concrete form of the above framework will be studied. Without loss of generality, we focus on the two view formulation, i.e., M equals 2, which can be easily extended to the case when M is larger than 2. The hinge loss is used to define $l(\mathbf{x}^{(p)}, y, \mathbf{w}^{(p)})$, and the squared loss is used for $l_c(\mathbf{x}^{(p)}, \mathbf{x}^{(q)}, \mathbf{w}^{(p)}, \mathbf{w}^{(q)})$. 2-norm is used as the regularization term $\Omega(\cdot)$. Then, the concrete form of the Eq.(1) turns to:

$$\begin{aligned} & \min_{\mathbf{w}^{(1)}, \mathbf{w}^{(2)}} \sum_{p=1}^2 \frac{\gamma_p}{2} \|\mathbf{w}^{(p)}\|^2 + \sum_{p=1}^2 C_p \sum_{i=1}^n \beta_i \xi_i^{(p)} \\ & + C \left(\sum_{i=1}^n \beta_i \|\mathbf{w}^{(1)T} \mathbf{x}_i^{(1)} - \mathbf{w}^{(2)T} \mathbf{x}_i^{(2)}\|^2 \right. \\ & \left. + \sum_{i=1}^m \|\mathbf{w}^{(1)T} \mathbf{z}_i^{(1)} - \mathbf{w}^{(2)T} \mathbf{z}_i^{(2)}\|^2 \right) \\ & s.t. \quad \forall i \in \{1, 2, \dots, n\}, \\ & \quad y_i \mathbf{w}^{(1)T} \mathbf{x}_i^{(1)} \geq 1 - \xi_i^{(1)}, y_i \mathbf{w}^{(2)T} \mathbf{x}_i^{(2)} \geq 1 - \xi_i^{(2)}. \end{aligned} \quad (4)$$

Here in this proposed method, we have absorbed the scaling components. i.e., $\frac{1}{n}$ and $\frac{1}{m+n}$ into the trade-off parameters C_p and C respectively, for simplicity. The final classification function f :

Algorithm: Multi-View Transfer Learning with a Large Margin Approach (MVTL-LM)

Input:

1. Reweighting Ratios for Source Domain Examples: $\beta_i, i = \{1, 2, \dots, n\}$
 2. Optimization Parameters: γ_1 and γ_2 for regularizers, and the trade-off parameters C, C_1, C_2 in Eq.(5), $\epsilon = 0.001$.
 3. Source Domain Examples: $\{(\mathbf{x}_1^{(i)}, y_1), (\mathbf{x}_2^{(i)}, y_2), \dots, (\mathbf{x}_n^{(i)}, y_n)\}, i \in \{1, 2, \dots, M\}$.
 4. Target Domain Examples: $\{\mathbf{z}_1^{(i)}, \mathbf{z}_2^{(i)}, \dots, \mathbf{z}_m^{(i)}\}, i \in \{1, 2, \dots, M\}$
- Output:**
The label assignment $\mathbf{l} = [l_1, l_2, \dots, l_m]$ for the target domain examples.

1. Initialization $t = 0$, randomly initialize $\tilde{\mathbf{w}}_0$.
2. Construct $\tilde{\mathbf{X}}^-, \tilde{\mathbf{X}}^{(1)}, \tilde{\mathbf{X}}^{(2)}$ according to Eq.(5).
3. Construct the matrix $\mathbf{H}(\beta, C)$.
4. repeat
5. $t = t + 1$
6. Compute the gradient for the empirical loss: $\mathbf{a}_t = \partial_{\tilde{\mathbf{w}}} R_{emp}(\tilde{\mathbf{w}}_{t-1})$, and $b_t = R_{emp}(\tilde{\mathbf{w}}_{t-1}) - \langle \tilde{\mathbf{w}}_{t-1}, \mathbf{a}_t \rangle$.
7. Derive the optimization problem: $R_t^{CP} = \max_{1 \leq i \leq t} \langle \tilde{\mathbf{w}}, \mathbf{a}_i \rangle + b_i$
8. $\tilde{\mathbf{w}}_t = \arg \min_{\tilde{\mathbf{w}}} \tilde{\mathbf{w}}^T \mathbf{H}(\beta, C) \tilde{\mathbf{w}} + R_t^{CP}$
9. $\epsilon_t = \min_{0 \leq i \leq t} J(\tilde{\mathbf{w}}_i) - J_t(\tilde{\mathbf{w}}_t)$
10. until $\epsilon_t \leq \epsilon$
11. **Classification Assignment:** for target domain example \mathbf{z}_i , if $0.5 \times \tilde{\mathbf{w}}^T (\tilde{\mathbf{z}}_i^{(1)} + \tilde{\mathbf{z}}_i^{(2)}) > 0$, l_i equals 1, and otherwise it equals -1 .

Table 1: Algorithm Description: Multi-View Transfer Learning with a Large Margin Approach (MVTL-LM)

$(\mathcal{X}^{(1)}, \mathcal{X}^{(2)}) \mapsto \mathbf{R}$ can be specified as: $f(\mathbf{x}) = \frac{\mathbf{w}^{(1)T} \mathbf{x}^{(1)} + \mathbf{w}^{(2)T} \mathbf{x}^{(2)}}{2}$. To simplify this formulation, several concatenate vectors are further introduced:

$$\begin{aligned} \tilde{\mathbf{w}} &= [\mathbf{w}_1^T, \mathbf{w}_2^T]^T, \tilde{\mathbf{x}}_i^{-T} = [\mathbf{x}_i^{(1)T}, -\mathbf{x}_i^{(2)T}]^T \\ \tilde{\mathbf{x}}_i^{(1)} &= [\mathbf{x}_i^{(1)T}, \mathbf{0}]^T, \tilde{\mathbf{x}}_i^{(2)} = [\mathbf{0}, \mathbf{x}_i^{(2)T}]^T, \end{aligned} \quad (5)$$

where in $\tilde{\mathbf{x}}_i^{(p)}$, only the $d_{(p-1)} + 1$ to d_p -th elements ($d_0 = 0$) are nonzero and equals $\mathbf{x}_i^{(p)}$. After introducing these notations, Eq.(4) can be simplified to the following form:

$$\begin{aligned} & \min_{\tilde{\mathbf{w}}} \tilde{\mathbf{w}}^T \mathbf{H}(\beta, C) \tilde{\mathbf{w}} + \sum_{p=1}^2 \sum_{i=1}^n C_p \beta_i \xi_i^{(p)} \\ & s.t. \quad \forall i \in \{1, 2, \dots, n\} \\ & \quad y_i \tilde{\mathbf{w}}^T \tilde{\mathbf{x}}_i^{(1)} \geq 1 - \xi_i^{(1)}, y_i \tilde{\mathbf{w}}^T \tilde{\mathbf{x}}_i^{(2)} \geq 1 - \xi_i^{(2)}, \end{aligned}$$

where $\mathbf{H}(\beta, C) = \mathbf{I}(\gamma_1, \gamma_2) + C(\sum_{i=1}^n \beta_i \tilde{\mathbf{x}}_i^- \tilde{\mathbf{x}}_i^{-T} + \sum_{i=1}^m \tilde{\mathbf{z}}_i^- \tilde{\mathbf{z}}_i^{-T})$, and $\mathbf{I}(\gamma_1, \gamma_2)$ is a diagonal matrix, with the first d_1 elements being $\frac{\gamma_1}{2}$ and the remaining ones being $\frac{\gamma_2}{2}$. It is clear that this problem is convex, since β_i is given.

There are several alternatives to solve this problem efficiently. Here, an efficient way, which is an adaption of the bundle method, is proposed to solve this optimization problem of the MVTL-LM approach. The concrete procedure is described in Table 1. Here, $R_{emp}(\tilde{\mathbf{w}}) = \sum_{p=1}^2 \sum_{i=1}^n C_p \beta_i \max\{0, 1 - y_i \tilde{\mathbf{w}}^T \tilde{\mathbf{x}}_i^{(p)}\}$, $J(\tilde{\mathbf{w}}) = \tilde{\mathbf{w}}^T \mathbf{H}(\beta, C) \tilde{\mathbf{w}} + \sum_{p=1}^2 \sum_{i=1}^n C_p \beta_i \xi_i^{(p)}$ $J_t(\tilde{\mathbf{w}}) = \tilde{\mathbf{w}}^T \mathbf{H}(\beta, C) \tilde{\mathbf{w}} +$

$\max_{1 \leq i \leq t} \langle \tilde{\mathbf{w}}, \mathbf{a}_i \rangle + b_i$. Since $R_{emp}(\tilde{\mathbf{w}})$ is non-smooth, so, when calculating its gradient, we use the sub-gradient instead, which can be calculated as

$$\partial_{\tilde{\mathbf{w}}} R_{emp}(\tilde{\mathbf{w}}) = - \sum_{p=1}^2 \sum_{i=1}^n C_p \beta_i I_i^{(p)} y_i^S \tilde{\mathbf{x}}_i^{(p)},$$

where $I_i^{(p)}$ is set to be 1, if $y_i^S \tilde{\mathbf{w}}^T \tilde{\mathbf{x}}_i^{(p)} < 1$, and $I_i^{(p)}$ is set to be 0, otherwise.

3.4 Theoretical Analysis

In this section, we deduct the convergence rate, time complexity as well as the generalized error bound of the proposed method.

3.4.1 Convergence

THEOREM 3.1. *For the convergence rate of the algorithm described in Table 1, Suppose $R_{max} = \max_{p,i} (C_p \beta_i) \|\tilde{\mathbf{x}}_i^{(p)}\|$, $\mathbf{A} = C(\sum_{i=1}^n \beta_i \tilde{\mathbf{x}}_i \tilde{\mathbf{x}}_i^T + \sum_{i=1}^m \tilde{\mathbf{z}}_i \tilde{\mathbf{z}}_i^T)$, the corresponding eigenvalues of \mathbf{A} are specified as: $\sigma_1(\mathbf{A}) \geq \sigma_2(\mathbf{A}) \geq \dots \geq \sigma_{(d_1+d_2)}(\mathbf{A}) \geq 0$. Assume that $\beta_i \leq B$. The proposed method converges in $O(1/\epsilon)$. In particular,*

- If $\epsilon > 16R_{max}^2 / (\min\{\gamma_1, \gamma_2\} + 2\sigma_{(d_1+d_2)}(\mathbf{A}))$, the proposed method converges to precision ϵ after at most $\log_2 \frac{nB(C_1+C_2)(\min\{\gamma_1, \gamma_2\} + 2\sigma_{(d_1+d_2)}(\mathbf{A}))}{4R_{max}^2}$ steps.
- If $\epsilon \leq 16R_{max}^2 / (\min\{\gamma_1, \gamma_2\} + 2\sigma_{(d_1+d_2)}(\mathbf{A}))$, the proposed method converges to precision ϵ after at most $\log_2 \frac{nB(C_1+C_2)(\min\{\gamma_1, \gamma_2\} + 2\sigma_{(d_1+d_2)}(\mathbf{A}))}{4R_{max}^2} + 32R_{max}^2 / (\epsilon(\min\{\gamma_1, \gamma_2\} + 2\sigma_{(d_1+d_2)}(\mathbf{A}))) - 1$ steps.

PROOF. We have $J(\tilde{\mathbf{w}}) = \tilde{\mathbf{w}}^T \mathbf{H}(\beta, C) \tilde{\mathbf{w}} + \sum_{p=1}^2 \sum_{i=1}^n C_p \beta_i \xi_i^{(p)}$, $\Omega(\tilde{\mathbf{w}}) = \tilde{\mathbf{w}}^T \mathbf{H}(\beta, C) \tilde{\mathbf{w}}$. $\Omega^*(\mu) = \mu^T \mathbf{H}^{-1}(\beta, C) \mu$ is the Fenchel dual of $\Omega(\tilde{\mathbf{w}})$. $J(\mathbf{0}) = \sum_{p=1}^2 \sum_{i=1}^n C_p \beta_i \leq nB(C_1 + C_2)$. $\|\partial_{\tilde{\mathbf{w}}} R_{emp}(\tilde{\mathbf{w}})\| \leq R_{max}$. It is clear that $\|\partial_{\mu}^2 \Omega^*(\mu)\| \leq 4 / (\min\{\gamma_1, \gamma_2\} + 2\sigma_{(d_1+d_2)}(\mathbf{A}))$. By integrating these inequalities into Theorem 4 of [39], we can get

$$\epsilon_t - \epsilon_{t+1} \geq \frac{\epsilon_t}{2} \min(1, \epsilon_t (\min\{\gamma_1, \gamma_2\} + 2\sigma_{(d_1+d_2)}(\mathbf{A})) / 16R_{max}^2),$$

where $\epsilon_t = \min_{0 \leq i \leq t} J(\tilde{\mathbf{w}}_i) - J_t(\tilde{\mathbf{w}}_i)$. The algorithm will terminate if $\epsilon_t < \epsilon$. So, if $\epsilon > 16R_{max}^2 / (\min\{\gamma_1, \gamma_2\} + 2\sigma_{(d_1+d_2)}(\mathbf{A}))$, $\epsilon_t - \epsilon_{t+1} \geq \frac{\epsilon_t}{2}$, and the algorithm will terminate after at most:

$$\begin{aligned} & \log_2 \frac{J(\mathbf{0})(\min\{\gamma_1, \gamma_2\} + 2\sigma_{(d_1+d_2)}(\mathbf{A}))}{4R_{max}^2} \\ & \leq \log_2 \frac{nB(C_1 + C_2)(\min\{\gamma_1, \gamma_2\} + 2\sigma_{(d_1+d_2)}(\mathbf{A}))}{4R_{max}^2} \end{aligned} \quad (6)$$

steps.

If $\epsilon \leq 16R_{max}^2 / (\min\{\gamma_1, \gamma_2\} + 2\sigma_{(d_1+d_2)}(\mathbf{A}))$, then, this method needs the above indicated steps to converge to the precision $16R_{max}^2 / (\min\{\gamma_1, \gamma_2\} + 2\sigma_{(d_1+d_2)}(\mathbf{A}))$, then, we should have $\epsilon_t - \epsilon_{t+1} \geq \frac{\epsilon_t^2}{2} (\min\{\gamma_1, \gamma_2\} + 2\sigma_{(d_1+d_2)}(\mathbf{A})) / 16R_{max}^2$. It is clear that it needs another $32R_{max}^2 / (\epsilon(\min\{\gamma_1, \gamma_2\} + 2\sigma_{(d_1+d_2)}(\mathbf{A}))) - 1$ steps to converge to the precision ϵ . So, in total, this algorithm converges in

$$\log_2 \frac{nB(C_1 + C_2)(\min\{\gamma_1, \gamma_2\} + 2\sigma_{(d_1+d_2)}(\mathbf{A}))}{4R_{max}^2}$$

$$+ 32R_{max}^2 / (\epsilon(\min\{\gamma_1, \gamma_2\} + 2\sigma_{(d_1+d_2)}(\mathbf{A}))) - 1$$

steps.

In summary, the algorithm converges in $O(1/\epsilon)$ steps. It is clear that the number of iterations also highly depends on R_{max}^2 , which can be viewed as the maximum reweighted norm for the source domain examples. Furthermore, it is clear that increasing the parameter values of C_1 and C_2 will increase the number of iterations. \square

3.4.2 Time Complexity

THEOREM 3.2. *For each iteration of the proposed method, it takes time $O(sn)$.*

PROOF. The gradient computation in step 6 takes time $O(sn)$, where s is the average sparsity on both views. Instead of solving the primal quadratic program, one can instead solve the optimization problem in step 8 in the dual form. Setting up the dual for each iteration is dominated by computing the $O(t^2)$ elements of the Hessian, which can be done in $O(t^2 s)$ steps. Since t^2 is normally much smaller than n , it leads to an overall time complexity of $O(sn)$ per iteration. \square

This result is similar to the time complexity result per iteration in [16]. However, the total number of iterations in [16] may be as worse as $O(1/\epsilon^2)$, as given by the Lemma 2 of [16]. On the contrary, the number of iterations required in this paper is guaranteed to be in $O(1/\epsilon)$. So, solving the proposed problem by the proposed method is much faster than using the Cutting Plane method [18].

3.4.3 Generalized Error Bound

In this subsection, we assume that $\forall \mathbf{x}, \beta(\mathbf{x}) \leq B$ if the marginal probabilities in the source domain or in the target domain are greater than 0. Let $\mathbf{w}_*^{(1)}$ and $\mathbf{w}_*^{(2)}$ denote the solution of Eq.(4). Similar to [12], we consider the class of functions $\mathcal{F}_{E,D} = \{f|f : \mathbf{x} \rightarrow \frac{1}{2}((\mathbf{w}^{(1)})^T \mathbf{x}^{(1)} + (\mathbf{w}^{(2)})^T \mathbf{x}^{(2)})\}$ such that $\|\mathbf{w}^{(1)}\|^2 \leq E^2$, $\|\mathbf{w}^{(2)}\|^2 \leq E^2$, and with probability at least $1 - \delta$,

$$\begin{aligned} & \frac{1}{m+n} \sum_{i=1}^n \beta_i \|(\mathbf{w}^{(1)})^T \mathbf{x}_i^{(1)} - (\mathbf{w}^{(2)})^T \mathbf{x}_i^{(2)}\|^2 \\ & + \frac{1}{m+n} \sum_{i=1}^m \|(\mathbf{w}^{(1)})^T \mathbf{z}_i^{(1)} - (\mathbf{w}^{(2)})^T \mathbf{z}_i^{(2)}\|^2 \\ & \leq \frac{1}{m+n} \sum_{i=1}^n \beta_i \|(\mathbf{w}_*^{(1)})^T \mathbf{x}_i^{(1)} - (\mathbf{w}_*^{(2)})^T \mathbf{x}_i^{(2)}\|^2 \\ & + \frac{1}{m+n} \sum_{i=1}^m \|(\mathbf{w}_*^{(1)})^T \mathbf{z}_i^{(1)} - (\mathbf{w}_*^{(2)})^T \mathbf{z}_i^{(2)}\|^2 \\ & + \frac{EB}{m+n} \sqrt{\sum_{i=1}^n \|\mathbf{x}_i\|^2 + \sum_{i=1}^m \|\mathbf{z}_i\|^2} + 3B \sqrt{\frac{\ln(2/\delta)}{2(m+n)}} =: D. \end{aligned}$$

Furthermore, define $\mathcal{F}_{\beta,E,D}$ to be the following class of functions,

$$\mathcal{F}_{\beta,E,D} = \{h|h : (\mathbf{x}, y) \rightarrow \beta(\mathbf{x}) \cdot \mathcal{A}(-f(\mathbf{x}) \cdot y), f \in \mathcal{F}_{E,D}\}$$

where $\mathcal{A}(a) = \begin{cases} 1 & \text{if } a > 0 \\ 1+a & \text{if } -1 \leq a \leq 0 \\ 0 & \text{otherwise} \end{cases}$. Based on these function classes, we have the following theorem with respect to the generalization error of the classifier obtained via the MVTL algorithm.

THEOREM 3.3. *Fix $\delta \in (0, 1)$. Then, with probability at least $1 - \delta$, every $f \in \mathcal{F}_{\beta,E,D}$ satisfies:*

$$\begin{aligned} & E_{P_{T}}(\text{sign}(f(\mathbf{x})) \neq y) \\ & \leq \frac{1}{n(C_1 + C_2)} \sum_{p=1}^2 C_p \sum_{i=1}^n \beta_i \xi_i^{(p)} + \hat{R}_{S,n}(\mathcal{F}_{\beta,E,D}) + 3B \sqrt{\frac{\ln(2/\delta)}{2n}}, \end{aligned}$$

where $\hat{R}_{S,n}(\mathcal{F}_{\beta,E,D})$ is the empirical Rademacher complexity of $\mathcal{F}_{\beta,E,D}$ in the source domain.

PROOF. Since the conditional probability of y given \mathbf{x} is the same for the source domain and the target domain,

$$E_{Pr_T}(\text{sign}(f(\mathbf{x})) \neq y) = E_{Pr_S}(\beta(\mathbf{x}) \cdot \text{sign}(f(\mathbf{x})) \neq y).$$

Notice that $\forall h \in \mathcal{F}_{\beta,E,D}$, $h(\mathbf{x}, y) \in [0, B]$. By similar proof as Theorem 4.17 in [33], we obtain that with probability greater than $1 - \delta$,

$$E_{Pr_S}(\beta(\mathbf{x}) \cdot \text{sign}(f(\mathbf{x})) \neq y) \leq E_{Pr_S}(\beta(\mathbf{x}) \cdot \mathcal{A}(-f(\mathbf{x}) \cdot y)) \\ \leq \frac{1}{n(C_1 + C_2)} \sum_{p=1}^2 C_p \sum_{i=1}^n \beta_i \xi_i^{(p)} + \hat{R}_{S,n}(\mathcal{F}_{\beta,E,D}) + 3B \sqrt{\frac{\ln(2/\delta)}{2n}}.$$

□

According to Theorem 3.3, the generalization error bound of any function in $\mathcal{F}_{\beta,E,D}$ depends on the weighted sum of all the slack variables associated with labeled examples in the source domain and the empirical Rademacher complexity of $\mathcal{F}_{\beta,E,D}$ in the source domain (up to a constant). Therefore, from transfer learning perspective, if the data distributions of the source domain and the target domain are similar (B is relatively small), constructing the large margin classifier in the source domain helps improve the generalization error of the classifier in the target domain. More importantly, from multi-view learning perspective, in $\mathcal{F}_{\beta,E,D}$, by leveraging the consistencies among different views, we effectively limit the hypothesis class, thus reduce the empirical Rademacher complexity. As empirically shown in [12], after imposing the consistency on different views, the Rademacher complexity is significantly reduced compared with the single-view correspondents.

This bound improves some existing theoretical results in transfer learning. For example, compared with the bound in Theorem 1 of [3], we make use of the data-dependent convergence measures, which can yield more accurate bounds. In Theorem 5 of [6], the authors also proved a bound for classifiers trained on multiple sources based on empirical Rademacher complexity. However, their bound can not be estimated from the data. For example, it depends on the expected difference between the labeling functions of different domains, which can *not* be estimated in our case where the target domain does *not* have any labeled examples.

Our bound also improves the results in [12], which is a multi-view learning algorithm. First, our bound is proposed for the transfer learning setting, whereas SVM-2K [12] is for the single domain setting. Furthermore, Theorem 3 of [12] can be seen as a special case of our bound when the source domain and the target domain have the same distribution and we do not use the unlabeled data from the target domain. Second, compared with SVM-2K, we make additional use of the unlabeled data from the target domain. If the number of unlabeled examples from the target domain is much larger than the number of labeled examples from the source domain, we tend to decrease the value of D , which is the upper bound on the view consistency. In this way, we reduce the function class, the corresponding Rademacher complexity, and thus improve the bound.

4. EXPERIMENTS

In this section, we present and analyze an extensive set of experimental results, which clearly demonstrate the advantages of the proposed method.

Dataset	Source Domain	Target Domain
comp vs rec	comp.graphics comp.os.ms-windows.misc rec.autos rec.motorcycles	comp.sys.ibm.pc.hardware comp.sys.mac.hardware rec.sport.baseball rec.sport.hockey
comp vs sci	comp.graphics comp.os.ms-windows.misc sci.electronics sci.space	comp.sys.ibm.pc.hardware comp.sys.mac.hardware sci.crypt sci.med
comp vs talk	comp.sys.ibm.pc.hardware comp.sys.mac.hardware talk.politics.guns talk.politics.mideast	comp.graphics comp.os.ms-windows.misc talk.politics.misc
rec vs sci	rec.autos rec.motorcycles sci.electronics sci.space	rec.sport.baseball rec.sport.hockey sci.crypt sci.med
rec vs talk	rec.sport.baseball rec.sport.hockey talk.politics.guns talk.politics.mideast	rec.autos rec.motorcycles talk.politics.misc
sci vs talk	sci.electronics sci.space talk.politics.guns talk.politics.mideast	sci.crypt sci.med talk.politics.misc

Table 2: Descriptions of Six Sub-datasets from 20Newsgroup

4.1 Datasets

4.1.1 20 Newsgroups

20 Newsgroups dataset¹ contains 4 main categories, i.e., ‘comp’, ‘rec’, ‘sci’, ‘talk’, as well as some small categories, such as ‘alt.atheism’, ‘misc.forsale’, etc. The number of examples for each of the four main categories ranges from 3253 to 4881. Each of the four main categories contains some subcategories, which are assigned to different domains. Using the 4 main categories, we create 6 sub-datasets. The detailed descriptions of these sub-datasets are summarized in Table 2. For each sub-dataset, we extract features from 2 views. One view (View 1) corresponds to the original tf-idf content information processed by Principle Component Analysis (PCA), and the other view (View 2) corresponds to the hidden topic information obtained by Probabilistic Latent Semantic Analysis (PLSA)² of the binary word features.

4.1.2 Spam Detection

This dataset is from ECML/PKDD Discovery Challenge 2006³, which focuses on personalized spam filtering and generalization across related learning tasks. In particular, in Task A, we aim to construct spam filters for 3 different users, each of which have 2500 emails. In our experiments, we take the labeled emails from one user as the source domain, and the unlabeled emails from another user as the target domain. The two views are generated using the same way as 20 Newsgroups.

4.1.3 WebKB

This dataset contains web pages from computer science departments of several different universities⁴. They are divided into 7

¹<http://people.csail.mit.edu/jrennie/20Newsgroups/>

²Actually, PLSA [14] can be considered as a dimensionality reduction method, which maps the documents to some fixed number of hidden topics. The topic distribution for each document can be used as low dimensional representation.

³<http://www.ecmlpkdd2006.org/challenge.html>

⁴<http://www.cs.cmu.edu/~webkb/>

	20 Newsgroups						WebKB		
	comp vs rec	comp vs sci	comp vs talk	rec vs sci	rec vs talk	sci vs talk	student	course	faculty
SVM-View1	85.4	73.3	95.4	67.2	56.8	81.8	0.595	0.541	0.441
SVM-View2	84.0	70.6	96.4	63.5	65.8	78.3	0.171	0.116	0.236
SVM	86.2	70.9	96.7	64.6	53.6	80.1	0.544	0.562	0.464
SVM-2K	88.5	79.3	97.2	71.0	81.6	83.1	0.725	0.512	0.563
CDSC	87.6	72.3	81.3	71.2	80.1	80.4	0.361	0.202	0.291
LLGC	77.5	71.1	92.6	72.6	73.9	81.2	0.167	0.277	0.310
LMTTL	85.8	69.8	96.4	64.5	53.7	78.6	0.546	0.545	0.475
Co-Training	86.9	72.5	97.1	66.7	61.2	81.4	0.495	0.554	0.444
MVTL-LM	92.9	80.1	98.3	75.3	83.4	83.0	0.742	0.565	0.671

Table 3: Classification Results on 20 Newsgroups and WebKB. For 20 Newsgroups, we report the classification accuracy; for WebKB, we report the F-measure due to the extremely imbalanced nature of this data set. It can be seen that MVTL-LM performs the best in most cases.

categories (i.e., student, faculty, staff, course, project, department and other). We generate three sub-datasets out of them, i.e., student, course and faculty. For each sub-dataset, we pick the corresponding webpages from the four main universities, i.e., Cornell, Washington, Wisconsin, and Texas as the source domain positive examples, and the webpages in 'other' category from these four universities as the source domain negative examples. In the target domain, a similar way is used to extract examples from the other universities. We use the content (View 1) and the link information (View 2) as the two views of this dataset.

4.2 Methods

We compare the proposed method with the following competitors: Support Vector Machines (SVM), which is a supervised classification method; LLGC [46], which is a semi-supervised learning method; SVM-2K [12], which is a multi-view learning method; CDSC [21], LMTTL [29], which are transfer learning methods and have shown state-of-the-art performances. We also adapt the co-training [27] algorithm to work for MVTL problems as follows: we disregard the domain difference, put labeled examples from the source domain and unlabeled examples from the target domain together, and apply the co-training algorithm to construct a classifier for each view of the target domain via SVM. This is similar to the co-adaptation algorithm proposed in [40] except that besides generating the initial seed set, labeled examples from the source domain are also used to construct classifiers for the target domain. Notice that besides SVM-2K and the co-training algorithm, the other baseline methods only work in the single-view settings. Therefore, for the sake of comparison, we first represent each example using a single set of features by concatenating the features from different views, and then apply these methods on this single view. Furthermore, to better understand the benefits brought by the multi-view methods, we also apply SVM on each view and report the performance.

For our proposed method, the re-weighting factors $\beta_i, i = 1, \dots, n$ are learned by Gaussian Mixture Model, and the numbers of Gaussian components are both set to be 4. We will show later that the performance of MVTL-LM is very robust against small perturbations in the number of Gaussian components. We set $\gamma_1 = \gamma_2 = 1$, and tune the remaining three parameters (C_1, C_2 and C) through five fold cross validation on both the source domain and the target domain. The parameters of SVM-2K are also set in the same way, except that in SVM-2K we do not need to consider the number of Gaussian components. For LLGC, the RBF kernel is used, with its Gaussian variance being determined automatically by local scaling [43]. The parameters of CDSC, SVM, LMTTL and Co-Training are all set through five fold cross validation similarly.

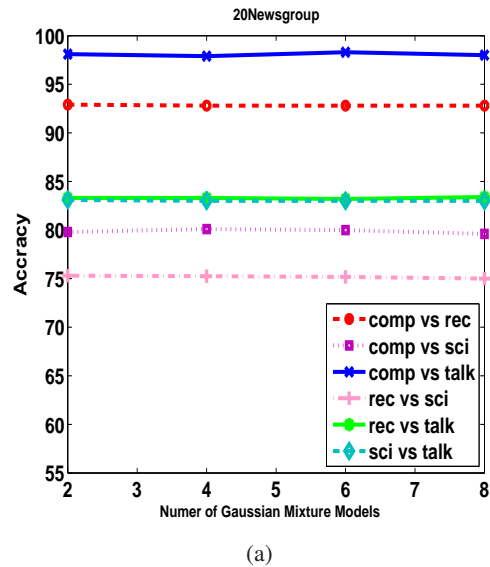


Figure 1: The impact of changing the number of Gaussian components on the performance of our proposed method. In this evaluation, we are assuming that the numbers of Gaussian components for both the source domain and the target domain are the same. The experiments are conducted by fixing the number of Gaussian components, while tuning the other parameters.

4.3 Results and Analysis

To study the impact of changing the number of Gaussian components on the performance of our proposed method, we vary this number when estimating β_i , and report the classification accuracy on 20 Newsgroups dataset in Fig.1. From these results, we can see that the proposed method is very stable with different numbers of Gaussian components. Similar results have also been observed on WebKB and Spam datasets.

Next we report the comparison results in Tables 3 and 4 respectively. For 20 Newsgroups and Spam datasets, the classification accuracy is reported; whereas for WebKB dataset, the F-measure is reported instead due to the extremely imbalanced nature of this dataset⁵. From these results, we have the following observations.

1. Our proposed method MVTL-LM performs the best in most cases. This is because our method models both the consis-

⁵The number of negative examples is around 6 times more than that of the positive examples.

	user1 vs user2	user1 vs user3	user2 vs user3	user2 vs user1	user3 vs user1	user3 vs user2
SVM-View1	79.7	65.7	83.4	76.7	76.0	80.9
SVM-View2	94.8	97.3	97.4	91.4	89.8	94.2
SVM	94.6	96.9	97.6	92.1	88.7	92.9
SVM-2K	94.1	97.6	97.2	91.8	90.8	94.1
CDSC	84.1	97.2	96.6	90.3	89.1	91.7
LLGC	97.1	96.3	93.2	91.7	91.3	93.2
LMTTL	94.9	96.2	97.5	92.0	88.9	93.1
Co-Training	92.5	96.6	96.6	92.2	88.6	93.9
MVTL-LM	95.2	97.9	98.1	93.7	92.9	96.3

Table 4: Classification Results on Spam dataset. The classification accuracy is reported. It can be seen that MVTL-LM performs the best in most cases.

tency between different views and the domain difference simultaneously, whereas the other methods ignore some useful information (i.e., the data distribution difference between different domains and the redundancy incurred by multiple views).

- Comparing with multi-view learning methods (SVM-2K and Co-Training), MVTL-LM performs better because it explicitly models the data distribution difference between the source domain and the target domain; whereas the multi-view learning methods simply treat the two domains as a single one.
- Comparing with transfer learning methods (CDSC and LMTTL), in MVTL-LM, we are able to transfer additional information about view consistency from the source domain to the target domain. We also suspect that these traditional transfer learning methods may not work well in the cases when different kinds of features are merged together. Therefore, MVTL-LM could achieve better performance in most cases.
- The performance of SVM is worse than SVM-2K in most cases. This is because in SVM, simply concatenating the features from different views together fails to capture the consistency between different views; whereas SVM-2K explicitly models this consistency, which is able to improve the overall performance.
- Comparing with SVM, SVM-View1, and SVM-View2, we can see that concatenating the features from different views may not necessarily result in an increase in the classification performances although SVM uses more information than SVM-View1 and SVM-View2.
- As a graph based semi-supervised method, the performance of LLGC is not promising because the basic manifold assumption in semi-supervised learning does not hold in transfer learning, and concatenating multi-view features together is not well suited in the multi-view learning scenario.
- The traditional transfer learning methods show very poor performances in WebKB. This is because in the subdatasets of WebKB, the link view contains too much noise as can be seen from the performance of SVM-View2. Concatenating these features together may bring more noise for classification, and therefore could cause a decrease in the classification performance, especially for traditional transfer learning methods.

These observations clearly demonstrate the advantages of the proposed method over state-of-the-art ones. It validates our claims and theoretical analysis that by integrating the multi-view learning and transfer learning together, the classification performance can be greatly improved.

5. CONCLUSIONS

Transfer learning is an important technique for utilizing data in a related source domain for building predictive models in a target domain. Much valuable prior research has been conducted for traditional transfer learning with data from a single view. However, many real world applications often contain data from multiple views, and there is limited work for transfer learning with data from different views. This paper proposes a formal learning framework for Multi-View Transfer Learning with a Large Margin (MVTL-LM) approach. In particular, the weighted labeled data from the source domain is used to construct a large margin classifier for target domain and both the unlabeled data from the target domain and data from source domain are used to ensure the classification consistency between different views. A novel optimization method based on bundle method is proposed to learn model parameters in an efficient manner, which has a theoretical guarantee to generate ϵ -accurate results in $O(1/\epsilon)$ steps. Furthermore, theoretical analysis is provided for the generalization error bound of the proposed method and shows the improved results of the Rademacher complexity. An extensive set of results on three different datasets have been provided to demonstrate the advantages of the proposed method against several other alternatives.

6. ACKNOWLEDGEMENT

We would like to express our sincere thanks to Dr. Zheng Wang (Tsinghua University), Prof. S.V.N. Vishwanathan (Purdue University), and the anonymous reviewers for their valuable comments and suggestions. This research was partially supported by the NSF research grants IIS-0746830, CNS-1012208, IIS-1017837, CCF-0939370.

7. REFERENCES

- E. C. Anderson. Monte Carlo Methods and Importance Sampling, 1999.
- A. Argyriou, T. Evgeniou, and M. Pontil. Multi-task feature learning. In *NIPS*, page 41. MIT Press, 2007.
- S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira. Analysis of representations for domain adaptation. In *NIPS*, pages 137–144, 2006.
- A. Blum and T. M. Mitchell. Combining labeled and unlabeled data with co-training. In *COLT*, pages 92–100, 1998.
- E. Bonilla, K. Chai, and C. Williams. Multi-task Gaussian process prediction. *NIPS*, 20:153–160, 2008.
- K. Crammer, M. Kearns, and J. Wortman. Learning from multiple sources. *Journal of Machine Learning Research*, 9:1757–1774, 2008.

- [7] W. Dai, G.-R. Xue, Q. Yang, and Y. Yu. Co-clustering based classification for out-of-domain documents. In *KDD*, pages 210–219, 2007.
- [8] W. Dai, Q. Yang, G. Xue, and Y. Yu. Boosting for transfer learning. In *ICML*, pages 200–207, 2007.
- [9] W. Dai, Q. Yang, G. Xue, and Y. Yu. Self-taught clustering. In *ICML*, pages 200–207, 2008.
- [10] J. Davis and P. Domingos. Deep transfer via second-order Markov logic. In *ICML*, pages 217–224, 2009.
- [11] L. Duan, I. Tsang, D. Xu, and S. Maybank. Domain transfer svm for video concept detection. *CVPR*, pages 1375–1381, 2009.
- [12] J. Farquhar, D. Hardoon, H. Meng, J. Shawe-Taylor, and S. Szedmak. Two view learning: SVM-2K, theory and practice. *NIPS*, 18:355, 2006.
- [13] K. Ganchev, J. Graça, J. Blitzer, and B. Taskar. Multi-view learning over structured and non-identical outputs. In *UAI*, pages 204–211, 2008.
- [14] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR*, pages 50–57, 1999.
- [15] J. Huang, A. Smola, A. Gretton, K. Borgwardt, and B. Scholkopf. Correcting sample selection bias by unlabeled data. *NIPS*, 19:601, 2007.
- [16] T. Joachims. Training linear SVMs in linear time. In *KDD*, pages 217–226, 2006.
- [17] T. Joachims and N. Cristianini. Composite kernels for hypertext categorisation. In *ICML*, pages 250–257, 2001.
- [18] J. Kelley. The cutting plane method for solving convex programs. *Journal of the SIAM*, 8(4):703–712, 1960.
- [19] N. Lawrence and J. Platt. Learning to learn with the informative vector machine. In *ICML*, page 65, 2004.
- [20] G. Li, S. C. H. Hoi, and K. Chang. Two-view transductive support vector machines. In *SDM*, pages 235–244, 2010.
- [21] X. Ling, W. Dai, G.-R. Xue, Q. Yang, and Y. Yu. Spectral domain-transfer learning. In *KDD*, pages 488–496, 2008.
- [22] X. Liu, Y. Gong, W. Xu, and S. Zhu. Document clustering with cluster refinement and model selection capabilities. In *SIGIR*, pages 191–198, 2002.
- [23] Y. Mansour, M. Mohri, and A. Rostamizadeh. Domain adaptation with multiple sources. In *NIPS*, pages 1041–1048, 2008.
- [24] Y. Mansour, M. Mohri, and A. Rostamizadeh. Multiple source adaptation and the rényi divergence. In *UAI*, pages 367–374. AUAI Press, 2009.
- [25] L. Mihalkova, T. Huynh, and R. Mooney. Mapping and revising Markov logic networks for transfer learning. In *AAAI*, volume 22, pages 608–613, 2007.
- [26] L. Mihalkova and R. Mooney. Transfer learning by mapping with minimal target data. In *Proceedings of the AAAI-08 Workshop on Transfer Learning for Complex Tasks*, 2008.
- [27] K. Nigam and R. Ghani. Analyzing the effectiveness and applicability of co-training. In *CIKM*, pages 86–93, 2000.
- [28] S. Pan, J. Kwok, and Q. Yang. Transfer learning via dimensionality reduction. In *Proceedings of AAAI*, pages 677–682.
- [29] B. Quanz and J. Huan. Large margin transductive transfer learning. In *CIKM*, pages 1327–1336, 2009.
- [30] R. Raina, A. Battle, H. Lee, B. Packer, and A. Ng. Self-taught learning: transfer learning from unlabeled data. In *ICML*, pages 766–763, 2007.
- [31] D. Rosenberg, V. Sindhwani, P. Bartlett, and P. Niyogi. A Kernel for Semi-Supervised Learning With Multi-View Point Cloud Regularization. *IEEE Signal Processing Magazine*, 2009.
- [32] B. Scholkopf and A. Smola. *Learning with kernels*. MIT press Cambridge, Mass, 2002.
- [33] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, New York, NY, USA, 2004.
- [34] S. Sheather and M. Jones. A reliable data-based bandwidth selection method for kernel density estimation. *Journal of the Royal Statistical Society. Series B (Methodological)*, 53(3):683–690, 1991.
- [35] H. Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000.
- [36] V. Sindhwani and P. Niyogi. A co-regularized approach to semi-supervised learning with multiple views. In *ICML Workshop on Learning with Multiple Views*, 2005.
- [37] V. Sindhwani and D. Rosenberg. An RKHS for multi-view learning and manifold co-regularization. In *ICML*, pages 976–983, 2008.
- [38] A. Smola, S. Vishwanathan, and Q. Le. Bundle methods for machine learning. *NIPS*, 20, 2008.
- [39] C. Teo, S. Vishwanathan, A. Smola, and Q. Le. Bundle methods for regularized risk minimization. *The Journal of Machine Learning Research*, 11:311–365, 2010.
- [40] G. Tur. Co-adaptation: Adaptive co-training for semi-supervised learning. In *ICASSP*, 2009.
- [41] M. Wu and B. Schölkopf. A local learning approach for clustering. In *NIPS*, pages 1529–1536, 2006.
- [42] B. Zadrozny. Learning and evaluating classifiers under sample selection bias. In *ICML*, 2004.
- [43] L. Zelnik-Manor and P. Perona. Self-tuning spectral clustering. *NIPS*, 17:1601–1608, 2004.
- [44] D. Zhang, F. Wang, C. Zhang, and T. Li. Multi-view local learning. In *AAAI*, pages 752–757, 2008.
- [45] T. Zhang, A. Popescul, and B. Dom. Linear prediction models with graph regularization for web-page categorization. In *SIGKDD*, pages 821–826, 2006.
- [46] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. In *NIPS*, 2003.
- [47] S. Zhu, K. Yu, Y. Chi, and Y. Gong. Combining content and link for classification using matrix factorization. In *SIGIR*, pages 487–494, 2007.