Collaborative Dual-PLSA: Mining Distinction and Commonality across Multiple Domains for Text Classification

Fuzhen Zhuang[1,2], Ping Luo[3], Zhiyong Shen[3], Qing He[1], Yuhong Xiong[4], Zhongzhi Shi[1], Hui Xiong[5]

[1] The Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, CAS, {zhuangfz, heq, shizz}@ics.ict.ac.cn
[2] Graduate University of Chinese Academy of Sciences
[3] Hewlett Packard Labs China, {ping.luo, zhiyongs}@hp.com
[4] Innovation Works, yhxiong@yahoo.com
[5] MSIS Department, Rutgers University, hxiong@rutgers.edu

ABSTRACT

The distribution difference among multiple data domains has been considered for the cross-domain text classification problem. In this study, we show two new observations along this line. First, the data distribution difference may come from the fact that different domains use different key words to express the same concept. Second, the association between this conceptual feature and the document class may be stable across domains. These two issues are actually the *distinction* and *commonality* across data domains.

Inspired by the above observations, we propose a generative statistical model, named Collaborative Dual-PLSA (CD-PLSA), to simultaneously capture both the domain distinction and commonality among multiple domains. Different from Probabilistic Latent Semantic Analysis (PLSA) with only one latent variable, the proposed model has two latent factors y and z, corresponding to word concept and document class respectively. The shared *commonality* intertwines with the *distinctions* over multiple domains, and is also used as the bridge for knowledge transformation. We exploit an Expectation Maximization (EM) algorithm to learn this model, and also propose its distributed version to handle the situation where the data domains are geographically separated from each other. Finally, we conduct extensive experiments over hundreds of classification tasks with multiple source domains and multiple target domains to validate the superiority of the proposed CD-PLSA model over existing state-of-the-art methods of supervised and transfer learning. In particular, we show that CD-PLSA is more tolerant of distribution differences.

Categories and Subject Descriptors

I.2.6 [**Artificial Intelligence**]: Learning–Machine Learning

General Terms

Algorithms, Experimentation, Performance.

Keywords

Statistical Generative Models, Cross-domain Learning, Classification

CIKM'10, October 26-30, 2010, Toronto, Ontario, Canada.

1. INTRODUCTION

Many classification techniques work well only under a common assumption that the training and test data are from the same data distribution. However, in many emerging real-world applications, new test data usually come from fast evolving information sources with different but semantically-related distributions. For example, to build an enterprize news portal we need to classify the news about a certain company into some predefined categories, such as "merger and acquisition", "product announcement", "financial scandal", and so on. This classification model may be trained from the news about one company, and may fail on the news for another company since the business areas for the two companies may be different. To deal with this change of data distributions, one solution is to include more labeled data in the new domains into the training set. However, it is often expensive or impractical to re-collect the needed training data, so reducing the need and the required effort to label new data is highly desired. This leads to the research of *cross-domain learning* (often referred to as *transfer learning* or *domain adaption*) [1, 2, 3, 4, 5, 6, 7, 8, 9].

Unlike previous work considering the distribution of the low-level features of raw words, we study high-level *word concepts*. Here, any word concept y can be represented by a multinomial distribution $p(w|y)$ over words, and this distribution is often domain-dependent. Let us take the word concept "products" as an example, if this concept is within the domain of the HP company, which makes printers, the values of $p(\text{"}printer\text{"}|\text{"}products\text{"})$ and $p(\text{"}LaserJet\text{"}|\text{"}products\text{"})$ are large within the domain of HP. If we change the domain to IBM, the representative words of this concept turn to be some IBM product names, and $p(\text{"}printer\text{"}|\text{"}products\text{"})$ and $p(\text{"}LaserJet\text{"}|\text{"}products\text{"})$ will have a very small value within the domain of IBM. Indeed, Table 4 in the experimental section also lists three word concepts with their key

words for each of the four domains. In the table, we can observe that different domains use different words to express and describe a concept.

Moreover, we observe that, wherever a word concept exists, it has the same implication to the class of the document which contains this concept. Let us consider the word concept "products". If a news contains the word concept "products", no matter where it comes from, it is more likely to be a news about "product announcement" rather than about "financial scandal". In other words, the association between word concept y and document class z, represented by their joint probability $p(y, z)$, is usually stable across domains.

In the above example, $p(w|y)$ and $p(y, z)$ corresponds to the two sides of a word concept y, *extension* and *intension* respectively. *In general, the extension of a concept is just the collection of individual objects to which it is correctly applied, while the intension of a concept is the set of features which are shared by everything to which it applies*[1]. Following the general definitions of concept extension and intension their definitions for word concept are as follows.

DEFINITION 1 (EXTENSION OF WORD CONCEPT). *The extension of a word concept y is the degree of applicability of that concept for each word w, denoted by $p(w|y)$.*

That is to say, when $p(w|y)$ is large, w is a typical object to which the word concept y can be applied.

DEFINITION 2 (INTENSION OF WORD CONCEPT). *The intension of a word concept y is expressed by its association with each document class z, denoted by their joint probability $p(y, z)$ in this study.*

For a word concept y, the values of $p(y, z)$ over different document classes z can be considered as the intrinsic features of concept y.

Similarly, we can also define the extension and intension of *document concept* z as $p(d|z)$ (a multinomial distribution over document d) and $p(y, z)$ respectively. Since we consider each document class for classification as a document concept here, document class and document concept are interchangeable in this paper.

With the above definitions, we further argue that the extension of any word concept or document concept is often domain-dependent, while its intension is often stable across different domains. Thus, the extension and intension of concepts are actually the distinction and commonality across data domains respectively. Motivated by this understanding, we propose a generative statistical model, Collaborative Dual-PLSA (CD-PLSA), to simultaneously capture both domain distinction and commonality. The main idea of this model is illustrated in Figure 1. In this figure, we have s source domains and t target domains (s and t can be any positive integers), represented by the dashed rectangle on the left and right respectively. In each dashed rectangle there are two solid rectangles at the above and below, bounding the extensions of word concepts and document concepts respectively. All these extensions, as the distinction for each domain, share the same intensions of word and document concepts as their commonality (the polygon in the middle). Since we know the class label of each document in the source domains, we actually know the extensions of the document

[1]http://www.philosophypages.com/lg/e05.htm

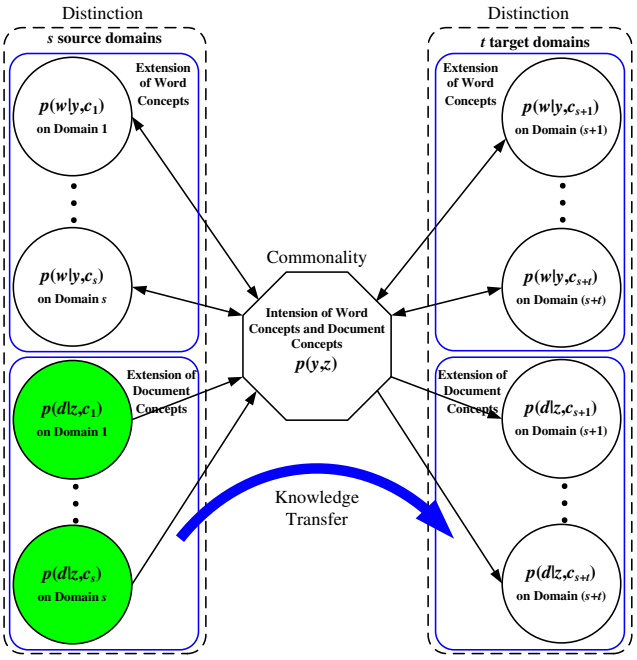


Figure 1: Extension and Intension of Concepts

concepts in the source domains. Thus, these observed extensions of the document concepts (the filled circles) are used as the supervision information, which is transferred through the bridge of concept intensions (the polygon in the middle) to the other parts of the model (the unfilled circles).

Contributions. In the following, we highlight some key contributions of this paper.

1) For the problem of text categorization across domains, we define the concepts of the extensions and intensions of words and documents, and show that concept extensions and intensions are actually the distinction and commonality across data domains.

2) We propose the generative model of CD-PLSA to mine the distinction and commonality of various data domains, and exploit an EM algorithm to learn the CD-PLSA model. Note that our model can simultaneously handle not only multiple source domains but also multiple target domains. To tackle the situation where the data domains are geographically separated from each other, we also provide a distributed solution to the CD-PLSA model.

3) Through comprehensive experiments, we show the effectiveness of the CD-PLSA model compared with the state-of-the-art methods. In particular, we clearly identify the scenarios where all the benchmark methods fail because the data distribution gap is too great to be handled, while the CD-PLSA model still performs well.

4) We further argue, contrary to popular belief that discriminative classifiers are always to be preferred, that generative classifiers (such as CD-PLSA proposed in this paper) may perform better in transfer learning since they can model the distribution differences among domains.

Overview. The remainder of this paper is organized as follows. In Section 2 we review some preliminaries and then give the problem formulation. Its solution by EM is followed in Section 3. Next, a distributed solution to CD-PLSA is described in Section 4 and the experimental results to validate

our algorithm are described in Section 5. Finally, the related works and conclusions are given in Sections 6 and 7.

2. PRELIMINARIES AND PROBLEM FORMULATION

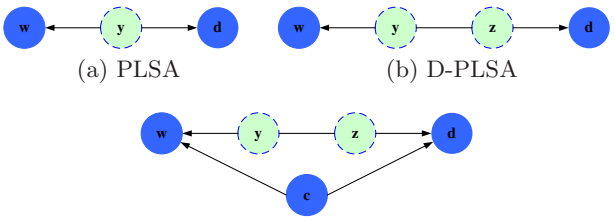
In this section, we first briefly review Probabilistic Latent Semantic Analysis (PLSA), and then introduce an extension of PLSA, Dual-PLSA. Finally, we formulate our problem for cross-domain classification.

2.1 A Review of PLSA

Probabilistic Latent Semantic Analysis [10] is a statistical model to analyze co-occurrence data by a mixture decomposition. Specifically, given the word-document co-occurrence matrix $\boldsymbol{O}$ whose element $O_{w,d}$ represents the frequency of word w appearing in document d, PLSA models $\boldsymbol{O}$ by using a mixture model with latent topics (each topic is denoted by y) as follows,

$$p(w,d) = \sum_y p(w,d,y) = \sum_y p(w|y)p(d|y)p(y). \quad (1)$$

Figure 2(a) shows the graphical model for PLSA. The parameters of $p(w|y), p(d|y), p(y)$ over all w, d, y are obtained by the EM solution to the maximum likelihood problem.



(a) PLSA (b) D-PLSA

(c) CD-PLSA

Figure 2: The Graphical Models

2.2 The Dual-PLSA Model

In the PLSA model, the documents and words share the same latent variable y. However, documents and words usually exhibit different organizations and structures. Specifically, they may have different kinds of latent topics, denoted by y for word concept and z for document concept. Its graphical model is shown in Figure 2(b). Since there are two latent variables in this model we call it Dual-PLSA (D-PLSA for short) in this paper.

Given the word-document co-occurrence $\boldsymbol{O}$, we can similarly arise a mixture model like Equation (1),

$$p(w,d) = \sum_{y,z} p(w,d,y,z) = \sum_{y,z} p(w|y)p(d|z)p(y,z). \quad (2)$$

And the parameters of $p(w|y), p(d|z), p(y,z)$ over all w, d, y, z can also be obtained by the EM solution. In these parameters $p(w|y)$ and $p(d|z)$ are actually the extensions of the word concept y and the document concept z respectively, while $p(y,z)$ is actually their intension.

This model was proposed in [11] for the clustering problem. In this paper we find that since the word topic and document topic are separated in this model we can inject the label information into $p(d|z)$ when d is a labeled instance and z is actually a document class. This way this model can also be used for semi-supervised classification. We will detail this in Section 5.1.2.

2.3 The Collaborative Dual-PLSA Model

Based on D-PLSA, we propose a statistical generative model for text classification cross multiple domains. Supposed we have $s+t$ data domains, denoted as $\mathcal{D} = (\mathcal{D}_1, \cdots, \mathcal{D}_s, \mathcal{D}_{s+1}, \cdots, \mathcal{D}_{s+t})$. Without loss of generality, we assume the first s domains are source domains with label information and the left t domains are target domains without any label information. Simply, for each domain we can generate its own extensions and intensions of word and document concepts. However, this simple method generates $s+t$ different sets of concept intensions. To obtain only one set of concept intensions, the variables y and z for word concept and document concept respectively must be independent of the variable c for the data domain. Therefore, we propose the graphical model in Figure 2(c) to catch the requirements that 1) y and z are independent of c; 2) the word w is dependent of both y and c; 3) the document d is dependent of both z and c. Given this graphical model the joint probability over all the variables is

$$p(w,d,y,z,c) = p(w|y,c)p(d|z,c)p(y,z)p(c). \quad (3)$$

The word-document co-occurrence matrix in the c-th domain is denoted by $\boldsymbol{O}_c$, whose element $O_{w,d,c}$ represents the co-occurrence frequencies of the triple (w,d,c). If we denote the two latent variables y, z as $\mathbf{Z}$, given the whole data $\boldsymbol{X}$ from different domains we formulate the problem of maximum log likelihood as

$$\log p(\boldsymbol{X}|\theta) = \log \sum_{\boldsymbol{Z}} p(\boldsymbol{Z}, \boldsymbol{X}|\theta), \quad (4)$$

where θ includes the parameters of $p(y,z)$, $p(w|y,c)$, $p(d|z,c)$ and $p(c)$.

We have to mention that although the extensions of the same word concept y on different domains are different, these extensions are semantically related to a certain degree. The reason is that they are trained collaboratively by sharing the same intension of $p(y,z)$. By the experimental results in Section 5.2.3 we will intuitively show the difference and relatedness among the extensions, which corresponds to the same word concept, on the multiple domains. In this sense we call our model Collaborative Dual-PLSA. Next, we develop an EM solution to the problem in Equation (4).

3. AN EM SOLUTION TO THE COLLABORATIVE DUAL-PLSA MODEL

An Expectation-Maximization (EM) algorithm is to maximize the lower bound (via Jensen's inequality) $\mathcal{L}_0$ of (4):

$$\mathcal{L}_0 = \sum_{\boldsymbol{Z}} q(\boldsymbol{Z}) \log \{\frac{p(\boldsymbol{Z}, \boldsymbol{X}|\theta)}{q(\boldsymbol{Z})}\}, \quad (5)$$

where $q(\boldsymbol{Z})$ could be arbitrary. We set $q(\boldsymbol{Z}) = p(\boldsymbol{Z}|\boldsymbol{X}; \theta^{\text{old}})$ and substitute into (5):

$$\begin{aligned} \mathcal{L}_0 &= \underbrace{\sum_{\boldsymbol{Z}} p(\boldsymbol{Z}|\boldsymbol{X}; \theta^{\text{old}}) \log p(\boldsymbol{Z}, \boldsymbol{X}|\theta)}_{\mathcal{L}} \\ &\quad \underbrace{- \sum_{\boldsymbol{Z}} p(\boldsymbol{Z}|\boldsymbol{X}; \theta^{\text{old}}) \log p(\boldsymbol{Z}|\boldsymbol{X}; \theta^{\text{old}})}_{\text{const}} \quad (6) \\ &= \mathcal{L} + \text{const}. \end{aligned}$$

3.1 E step: constructing $\mathcal{L}$

According to the derivation in Appendix, for the problem setting of CD-PLSA we have

$$\begin{aligned}\mathcal{L} &= \sum_{y,z,w,d,c} O_{w,d,c} p(y,z|w,d,c;\theta^{\text{old}}) \\ &\quad \cdot \log\,[p(y,z)p(w|y,c)p(d|z,c)p(c)],\end{aligned} \tag{7}$$

where

$$\begin{aligned}&p(y,z|w,d,c;\theta^{\text{old}}) \\ =&\frac{p(y,z)p(w|y,c)p(d|z,c)p(c)}{\sum_{y,z}\ p(y,z)p(w|y,c)p(d|z,c)p(c)}.\end{aligned} \tag{8}$$

3.2 M step: maximizing $\mathcal{L}$

Now we maximize $\mathcal{L}$ with its parameters by Lagrangian Multiplier method. Expand $\mathcal{L}$ and extract the terms containing $p(w|y,c)$. Then, we have $\mathcal{L}_{[p(w|y,c)]}$ and apply the constraint $\sum_w p(w|y,c) = 1$ into the following equation:

$$\frac{\partial\left[\mathcal{L}_{[p(w|y,c)]} + \lambda(\sum_w p(w|y,c) - 1)\right]}{\partial p(w|y,c)} = 0, \tag{9}$$

we have

$$\hat{p}(w|y,c) \propto \sum_{z,d} O_{w,d,c}\ p(y,z|w,d,c;\theta^{\text{old}}). \tag{10}$$

Note that we should normalize $\hat{p}(w|y,c)$ via

$$\hat{p}(w|y,c) = \frac{\sum_{z,d} O_{w,d,c}\ p(y,z|w,d,c;\theta^{\text{old}})}{\sum_{z,w,d} O_{w,d,c}\ p(y,z|w,d,c;\theta^{\text{old}})}. \tag{11}$$

Similarly,

$$\hat{p}(d|z,c) = \frac{\sum_{y,w} O_{w,d,c}\ p(y,z|w,d,c;\theta^{\text{old}})}{\sum_{y,w,d} O_{w,d,c}\ p(y,z|w,d,c;\theta^{\text{old}})}, \tag{12}$$

$$\hat{p}(y,z) = \frac{\sum_{w,d,c} O_{w,d,c}\ p(y,z|w,d,c;\theta^{\text{old}})}{\sum_{y,z,w,d,c} O_{w,d,c}\ p(y,z|w,d,c;\theta^{\text{old}})}, \tag{13}$$

$$\hat{p}(c) = \frac{\sum_{y,z,w,d} O_{w,d,c}\ p(y,z|w,d,c;\theta^{\text{old}})}{\sum_{y,z,w,d,c} O_{w,d,c}\ p(y,z|w,d,c;\theta^{\text{old}})}. \tag{14}$$

3.3 CD-PLSA to Cross-domain Classification

In this subsection, we introduce how to leverage the proposed EM algorithm for cross-domain classification. We need to figure out two sub-tasks: 1) how to inject the label information in source domains to supervise the EM optimization; 2) how to assign the class label to the instances in the target domains based on the output from the EM algorithm.

For the first task we inject the supervising information (the class label of the instances in the source domains) into the probability $p(d|z,c)$ $(1 \le c \le s)$. Specifically, let $\boldsymbol{L}^c \in [0,1]^{n_c \times m}$ be the true label information of the c-th domain, where n_c is the number of instances in it, m is the number of document classes. If instance d belongs to document class z_0, then $L^c_{d,z_0} = 1$, otherwise $L^c_{d,z} = 0$ $(z \neq z_0)$. We normalize $\boldsymbol{L}^c$ to satisfy the probabilistic condition so that the sum of the entries in each column equals to 1,

$$N^c_{d,z} = \frac{L^c_{d,z}}{\sum_d L^c_{d,z}}. \tag{15}$$

Then $p(d|z,c)$ is initialized as $N^c_{d,z}$. Note that since this initial value is from the true class label we do not change the value of $p(d|z,c)$ (for $1 \le c \le s$) during the iterative process.

For the unlabeled target domains, $p(d|z,c)$ $(s+1 \le c \le s+t)$ can be initialized similarly. This time the label information $\boldsymbol{L}^c$ used can be obtained by any supervised classifier (Logistic Regression is used in this paper). Note that since this classifier may output the wrong class label we do change the value of $p(d|z,c)$ (for $s+1 \le c \le s+t$) during the iterative process.

Algorithm 1 CD-PLSA for Cross-domain Classification

Input: Given $(s+t)$ data domains $\mathcal{D}_1, \cdots, \mathcal{D}_s, \mathcal{D}_{s+1}, \cdots, \mathcal{D}_{s+t}$, where the first s domains are source domains while the left are target domains. T, the number of iterations. Y, the number of word clusters.

Output: the class label of each document d in the target domain.

1. Initialization. $p^{(0)}(w|y,c)$ is set to the output $p(w|y)$ from PLSA. The initialization of $p^{(0)}(d|z,c)$ is detailed in Section 3.3. $p^{(0)}(y,z)$ is set randomly.
2. $k := 1$.
3. for $c := 1 \ \rightarrow s+t$

   Update $p^{(k)}(y,z|w,d,c)$ according to Equation (8) in **E-step**;
4. end.
5. for $c := 1 \ \rightarrow s+t$

   Update $p^{(k)}(w|y,c)$ according to Equation (11) in **M-step**;
6. end.
7. for $c := s+1 \ \rightarrow s+t$

   Update $p^{(k)}(d|z,c)$ according to Equation (12) in **M-step**;
8. end.
9. Update $p^{(k)}(y,z)$ according to Equation (13) in **M-step**.
10. Update $p^{(k)}(c)$ according to Equation (14) in **M-step**.
11. $k := k+1$, if $k < T$, turn to Step 3.
12. The class label of any document d in a target domain c is predicted by Equation (17).

After the EM iteration we obtain all the parameters of $p(d|z,c)$, $p(w|y,c)$, $p(y,z)$, $p(c)$, based on which we compute the posteriori probability $p(z|d,c)$ as follows,

$$\begin{aligned}p(z|d,c) &= \frac{p(z,d,c)}{p(d,c)} \propto p(z,d,c) = p(d|z,c)p(z,c) \\ &= p(d|z,c)p(z)p(c) = p(d|z,c)p(c)\sum_y p(y,z) \\ &\propto p(d|z,c)\sum_y p(y,z).\end{aligned} \tag{16}$$

Then, the class label of any document d in a target domain c is predicted to be

$$\arg\max_z p(z|d,c). \tag{17}$$

The detailed procedure of CD-PLSA for cross-domain classification is depicted in Algorithm 1. Note that our algorithm can deal with the situations there are multiple source domains and multiple target domains.

4. A DISTRIBUTED IMPLEMENTATION OF THE CD-PLSA MODEL

In this section we extend the proposed EM algorithm into a distributed version, which can work in the situation that the source domains $\mathcal{D}_1, \cdots, \mathcal{D}_s$ and the target domains $\mathcal{D}_{s+1}, \cdots, \mathcal{D}_{s+t}$ are geographically separated. This distributed computing helps when we cannot gather all the raw data from the separated data domains together due to security or other issues.

In this distributed setting, we need a cental node, denoted by mn, as the *master node*, and all the nodes for the data domains are used as *slave nodes*, denoted by $sn^{(1)}, \cdots, sn^{(s+t)}$. We find that 1) $p(y, z|w, d, c; \theta^{\text{old}})$, $p(w|y, c)$ and $p(d|z, c)$ in Equation (8), (11) and (12) can computed locally on $sn^{(c)}$; 2) $p(y, z)$ can be computed locally on the master node. Specifically, let

$$\triangle_{y,z}^{(c)} = \sum_{w,d} O_{w,d,c}\ p(y, z|w, d, c; \theta^{\text{old}}), \tag{18}$$

$$\mathcal{V}^{(c)} = \sum_{y,z,w,d} O_{w,d,c}\ p(y, z|w, d, c; \theta^{\text{old}}), \tag{19}$$

Then,

$$p(y, z) = \frac{\sum_c \triangle_{y,z}^{(c)}}{\sum_{y,z,c} \triangle_{y,z}^{(c)}}, \quad p(c) = \frac{\mathcal{V}^{(c)}}{\sum_c \mathcal{V}^{(c)}}. \tag{20}$$

In each iteration, the master node first sends the values of $p(y, z)$ and $p(c)$ to each slave node. Then, each slave node $sn^{(c)}$ (for $c \in \{1, \cdots, (s+t)\}$) computes $p(y, z|w, d, c; \theta^{\text{old}})$, $p(w|y, c)$, $p(d|z, c)$, $\triangle_{y,z}^{(c)}$ and $\mathcal{V}^{(c)}$ locally, and sends the local statistics $\triangle_{y,z}^{(c)}$ and $\mathcal{V}^{(c)}$ to the master node. Finally, the master node updates $p(y, z)$ and $p(c)$ according to Equation (20) when receiving all the local statistics from slave nodes, and starts the new round of iteration. It is clear that there are only some statistics, including $\triangle_{y,z}^{(c)}$, $p(y, z)$, $\mathcal{V}^{(c)}$ and $p(c)$, transmitted between the master and slave nodes (depicted in Figure 3), rather than communicating and exposing the raw domain data. Let T be the number of iterative rounds, Y be the number of word clusters, C be the number of document classes, then the total communication overhead are $2T \cdot (s + t) \cdot (Y \cdot C + 1)$ (the size of both $p(y, z)$ and $\triangle_{y,z}^{(c)}$ are $Y \cdot C$). Therefore, this distributed algorithm is communication-efficient and also alleviate the privacy concerns to some degree.

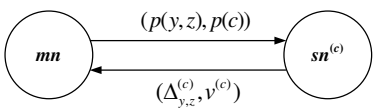


Figure 3: The statistics transmitted between the master and slave nodes.

5. EXPERIMENTS

In this section we design systemic experiments to demonstrate the effectiveness of CD-PLSA. In these experiments we focus on two-class classification problems, each of which involves with four domains: one source domain plus three target domains or three source domains plus one target domain. The classification accuracy is the evaluation metric in this work.

5.1 The Experimental Setup

5.1.1 Data Preparation

20-Newsgroup[2] is one of the widely used data set for cross-domain learning. This corpus has approximately 20,000 newsgroup documents, which are evenly divided into 20 subcategories. Some related subcategories are grouped into a top category, which is used as document class. Then we construct a cross-domain classification problem as follows. For two top categories A and B their four subcategories are denoted as A_1, A_2, A_3, A_4 and B_1, B_2, B_3, B_4, respectively. We select (without replacement) a subcategory from A (e.g., A_2) and a subcategory from B (e.g., B_3) to form a data domain. We repeat the selection four times to get the four data domains. Then, we select any one of the generated four domains as source domain and the left three domains as target domains. This way we can generate totally 96 ($4 \cdot P_4^4$) problems of cross-domain classification with one source domain and three target domains. Similarly, we can construct 96 problems with three source domains and one target domain. In the experiments we use three top categories *comp*, *rec* and *sci*. Their corresponding subcategories are listed in Table 1. The value of 15 is used as the threshold of document frequency to cut down the number of words used in the co-occurrence matrices.

Table 1: The top categories and their subcategories

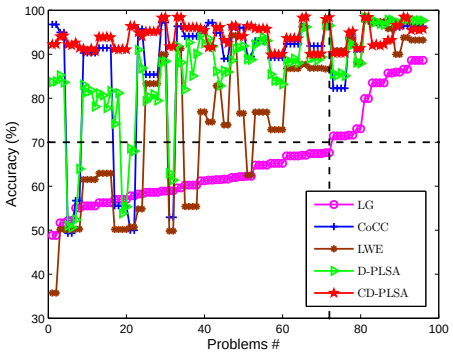
Top Categories	Subcategories
comp	*comp.*{*graphics, os.ms-windows.misc*} *comp.sys.*{*ibm.pc.hardware, mac.hardware*}
rec	*rec.*{*autos, motorcycles*} *rec.sport.*{*baseball, hockey*}
sci	*sci.*{*crypt, med, electronics, space*}

5.1.2 The Baseline Methods

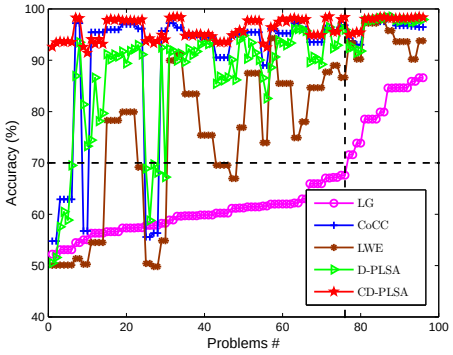
We compare CD-PLSA with several baseline classification methods, including the supervised learning algorithm Logistic Regression (LG) [12], and the cross-domain learning approaches Co-clustering based Classification (CoCC)[3] [5] and Local Weighted Ensemble (LWE) [2]. Since CoCC can not tackle the scenario with multiple source domains, we adapt the method of CoCC for handling m source domains as follows. For each source domain and the target domain we train a CoCC model, and then combine these m models by voting with equal weights. LG is adapted to deal with multiple source domains similarly with CoCC (Note that LG achieves the similar performance when trained on the merged data of all source domains). Additionally, the algorithm D-PLSA (depicted in Section 2.2) is also used as the baseline. Since there are not domain labels in D-PLSA all the instances appear as if they are from the same domain. In other words the source of each instance is ignored in D-PLSA. Our experiments will show that ignoring this information results in the significant performance sacrifice.

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

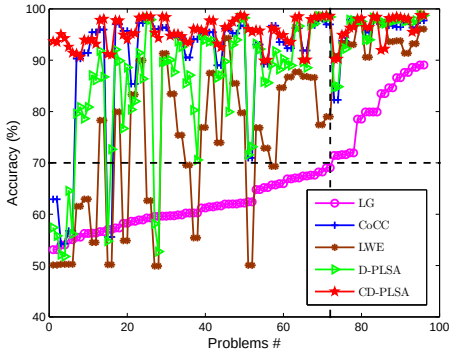
[3]We thank the author provides the codes.



(a) CD-PLSA vs. D-PLSA, LWE, CoCC, LG on Target Domain 1

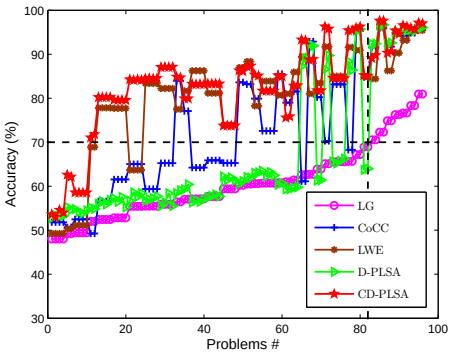


(b) CD-PLSA vs. D-PLSA, LWE, CoCC, LG on Target Domain 2

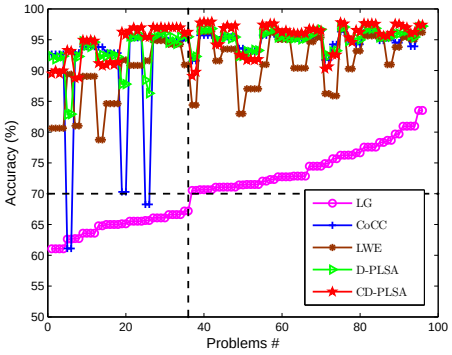


(c) CD-PLSA vs. D-PLSA, LWE, CoCC, LG on Target Domain 3

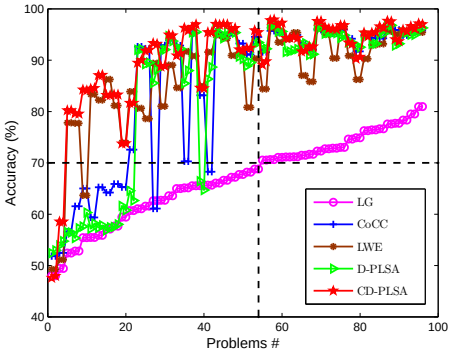
Figure 4: The Performance Comparison among CD-PLSA, D-PLSA, LWE, CoCC and LG on data set *rec vs. sci*



(a) CD-PLSA vs. D-PLSA, LWE, CoCC, LG on Target Domain 1



(b) CD-PLSA vs. D-PLSA, LWE, CoCC, LG on Target Domain 2



(c) CD-PLSA vs. D-PLSA, LWE, CoCC, LG on Target Domain 3

Figure 5: The Performance Comparison among CD-PLSA, D-PLSA, LWE, CoCC and LG on data set *comp vs. sci*

5.1.3 *Implementation Details*

Since the models of D-PLSA and CD-PLSA have the random initialization process, we conduct the experiments three times and the average results are recorded for these two algorithms. Preliminary test shows that our algorithm is not sensitive to the number Y of word clusters (in the range of $[2^5, 2^8]$), thus we set Y to 64. The number of iteration in both D-PLSA and CD-PLSA[4] is set to 50. The parameters of CoCC and LWE are set to the same values as those in their original papers.

[4]Under these parameters, CD-PLSA can finish our task in 240 seconds. Note that there are about 7300 features and 7500 documents in each problem.

5.2 Experimental Results

5.2.1 *Multiple Target Domains*

Here, we show a comparison of the proposed CD-PLSA model with the baseline methods on the learning tasks with multiple target domains. Since we have the data from the three top categories, we can select any two of them to construct 96 problems. Here, we only list the results from *rec vs. sci* and *comp vs. sci*. All the results are shown in Figures 4 and 5. Each of these two figures have three sub-figures, each

of which contains the results on one of the three target domains. In each sub-figure, the 96 problems are sorted by the increasing order of the accuracy from LG. Thus, the x-axis in each figure actually indicates the degree of difficulty in knowledge transformation. From these figures, we can observe that:

1) The t-test with 95% confidence over all the 192 (96×2) problems in Figures 4 and 5 shows that CD-PLSA significantly outperforms the other four baseline methods. Furthermore, we find that the improvements of CD-PLSA over the baseline methods are more remarkable when the accuracy of LG is lower than 70%. Table 2 records the average results over the corresponding tasks. The $Left$ and $Right$ rows represent the average values of the tasks when the accuracy of LG is lower or higher than 70% respectively, while $Total$ denotes the average values over all the 96 problems. You can see that the difference between the average values of CD-PLSA and any baseline method in the $Left$ row is much greater than that in the $Right$ row. That is to say, although the baseline methods may output much lower accuracies when the accuracy of LG is lower than 70%, CD-PLSA works still well. The reason may be that the degree of difference in data distributions across domains is too large to be handled by the baseline methods, while our method is more tolerant of distribution differences.

2) We also observe the advantage of CD-PLSA over D-PLSA in these results. The reasons are as follows. In D-PLSA, the data domain where each instance comes from is ignored, and all the instances are treated as if they come from the same domain. However, the distinction and commonality can only be found by the comparison of at least two domains. Thus, with only one domain our algorithm may sacrifice due to the loss of the information of data domains. On the other hand, we can say that the data domain for each instance introduces significant improvements to our model CD-PLSA.

5.2.2 Multiple Source Domains

Here, we conduct experiments to show that the CD-PLSA model can also work on multiple source domains. We evaluate all the methods on the problem with 3 sources and 1 target. Figure 6 shows the results. Indeed, similar results can be observed as those in Section 5.2.1, which again show that CD-PLSA outperforms all the compared methods.

We also show Table 3 with the average values over the corresponding 96 problems of the two data sets. The calculation of these values are the same with that in Table 2. Again, these results show that CD-PLSA outperforms the baseline methods on the tasks with multiple source domains, and it can better tolerate the distribution differences.

Table 3: Average Performances (%) on 96 Problems of Each Data Set for Multiple Source Domains

Data Sets		LG	CoCC	LWE	D-PLSA	CD-PLSA
rec vs. sci	*Left*	64.01	80.07	71.41	92.03	**94.06**
	Right	79.84	**97.70**	93.62	96.77	96.46
	Total	72.42	89.44	83.21	94.55	**95.33**
comp vs. sci	*Left*	60.15	74.88	76.77	63.21	**80.54**
	Right	79.91	**95.58**	92.14	94.38	94.51
	Total	74.97	90.41	88.30	86.59	**91.02**

5.2.3 Understanding the Extension of a Word concept over Multiple Domains

Here, we show the difference and relatedness among the extensions of a word concept over multiple domains. Fixing a word concept y and a domain c, we list the top N ($N = 20$ here) words in terms of $p(w|y, c)$. They are actually the representative words for the word concept in a certain domain. The extensions of three word concepts in the four domains are listed in Table 4.

Indeed, the extensions of a word concept on the four domains are related to each other in the sense that their representative words corresponds to the same *semantic*. For example, the third word concept is actually about "Space Science", while the representative words in each extension are different. Specifically, the representative words of this concept in Domain 1 include "rocket", "ESA" (European Space Agency), and "satellite" etc, while those in Domain 2 contain "acceleration", "NASA", and "earth" etc. These results also intuitively show that our model can successfully mine distinction and commonality among multiple domains.

5.3 Experimental Summary

We summary all the experimental results as follows:

1) CD-PLSA significantly outperforms all the baseline methods. This superiority becomes more remarkable when the accuracy from LG is lower than 70%. This indicates that, when the degree of difficulty in knowledge transfer is large, our model still works well while the others may fail. Thus, CD-PLSA is more robust for transfer learning.

2) The CD-PLSA model is further improved by explicitly considering the data domain where each instance comes from. Since the distinction and commonality can only be identified by the comparison of at least two domains, if all the instances are treated as if they come from the same domain, the effectiveness in mining distinction and commonality may compromise. Indeed, the data domain labels on each instance provide a partition of all the data if we group the instances from the same domain into one cluster. Thus, this information is additional supervision to our model.

3) To intuitively understand the extensions of a word concept over different domains, we list the key words of a concept in different domains. These key words, the bi-product of our model, coincide with our assumption that different domains use different words to describe the same concept.

6. RELATED WORKS AND DISCUSSIONS

In this section, we will survey some related work, and then give some discussions on generative and discriminative classifiers for cross-domain learning.

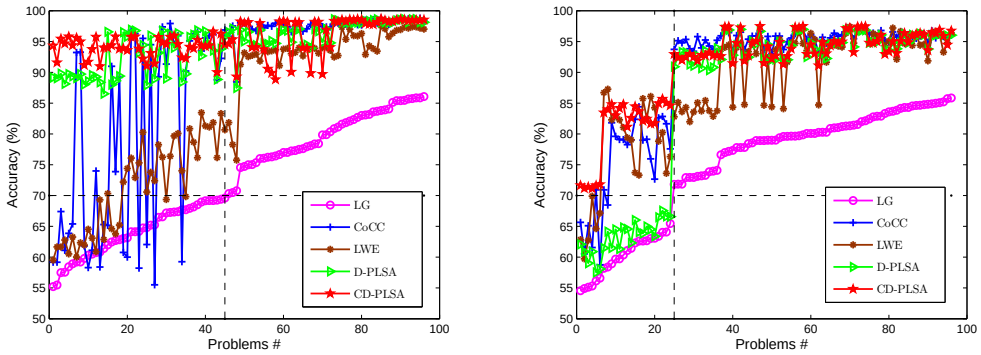
6.1 Cross-domain Learning

Cross-domain Learning has attracted great attention in recent years, and the works in this field can be grouped into four categories based on the different types of techniques used for knowledge transfer, namely feature selection based, feature space mapping, weight based, and model combination based methods.

Feature selection based methods are to identify the common features (at the level of raw words) between source and target domains, which are useful for transfer learning. Jiang et al. [13] argued that the features highly related to class labels should be assigned to large weights in the learnt model, thus they developed a two-step feature selection framework for domain adaptation. They first selected the general features to build a general classifier, and then considered the unlabeled target domain to select specific features for training target classifier. Zhuang et al. [14] formulated a joint optimization framework of the two matrix tri-factorizations for

Table 2: Average Performances (%) on 96 Problems of Each Data Set for Multiple Target Domains

Data Sets		Target-1					Target-2					Target-3				
		LG	CoCC	LWE	D-PLSA	CD-PLSA	LG	CoCC	LWE	D-PLSA	CD-PLSA	LG	CoCC	LWE	D-PLSA	CD-PLSA
rec vs. sci	*Left*	60.33	86.32	69.78	83.16	**93.98**	59.72	89.34	73.24	86.23	**95.82**	61.00	89.62	72.49	84.70	**95.39**
	Right	80.82	93.70	93.47	**94.31**	94.09	80.88	96.47	94.29	96.72	**97.64**	81.11	95.55	94.02	95.94	**96.32**
	Total	65.46	88.17	75.70	85.95	**94.00**	64.13	90.82	77.62	88.42	**96.20**	66.03	91.10	77.87	87.51	**95.62**
comp vs. sci	*Left*	57.93	69.10	77.64	62.54	**80.64**	64.66	89.52	88.44	92.71	**94.08**	61.02	77.30	82.05	76.86	**86.53**
	Right	75.70	94.14	91.56	**94.74**	94.54	74.70	94.95	92.36	95.29	**95.65**	74.36	94.64	92.65	94.30	**95.02**
	Total	60.52	72.75	79.67	67.23	**82.66**	70.93	92.91	90.89	94.33	**95.06**	66.86	84.89	86.68	84.49	**90.25**



(a) CD-PLSA vs. D-PLSA, LWE, CoCC, LG on data set *rec vs. sci*

(b) CD-PLSA vs. D-PLSA, LWE, CoCC, LG on data set *comp vs. sci*

Figure 6: The Performance Comparison among CD-PLSA, D-PLSA, LWE, CoCC and LG on two data sets

Table 4: Word concepts with their key words for each domain

Associated with Concept: *Space Science*	Domain 1	rocket, esa, assist, frank, af, thu, helsinki, ron, atlantic, jet, observer, satellite, venus, sei, min, ir, russia, stars, star, ray
	Domain 2	relay, km, rat, pixel, command, elements, arc, acceleration, nasa, earth, fuse, ground, bulletin, pub, anonymous, faq, unix, cit, ir, amplifier
	Domain 3	from, earth, science, word, pictures, years, center, data, national, dale, nasa, gif, reports, mil, planet, field, jpl, ron, smith, unix
	Domain 4	service, archive, unit, magnetic, thousands, technology, information, arc, keys, faq, probes, ir, available, gov, embedded, tens, data, system, unix, mil
Associated with Concept: *Computer Science*	Domain 1	support, astronomer, near, thousands, million, you, vnet, copy, ad, bright, lab, idea, data, hardware, engines, ibm, project, soviet, software, program
	Domain 2	legally, schemes, protected, bytes, mq, disks, patch, registers, machine, pirates, install, card, rom, screen, protection, disk, ram, tape, mb, copy
	Domain 3	discomfort, friend, normal, self, tests, programmer, steve, state, program, lab, you, your, jon, my, headache, trial, she, pain, page, trials
	Domain 4	wcs, cipher, scheme, brute, user, file, encryption, message, serial, decryption, crypto, keys, cryptosystems, skipjack, plaintext, secure, key, encrypted, nsa, des
Associated with Concept: *Car*	Domain 1	saves, power, was, at, disappointment, al, europeans, will, ny, north, their, they, deal, best, year, sports, cs, new, series, gm
	Domain 2	crash, price, vehicle, insurance, handling, gas, xs, dealer, cruiser, leather, buy, latech, fj, paint, ride, buying, bmw, engine, car, honda
	Domain 3	or, value, they, wade, good, car, better, best, three, performance, more, runner, than, average, dl, extra, base, cs, al, year
	Domain 4	dealer, camry, saab, engine, eliot, requests, mazda, liter, mustang, diesel, wagon, nissan, mileage, byte, saturn, toyota, si, cars, car, db

* Domain 1: *rec.sport.hockey* vs. *sci.space*, Domain 2: *rec.motorcycles* vs. *sci.electronics*
Domain 3: *rec.sport.baseball* vs. *sci.med*, Domain 4: *rec.autos* vs. *sci.crypt*.

the source and target domain data respectively, in which the associations between word clusters and document classes are shared between them for knowledge transfer. Although the basic assumption of this method is similar to our method, it lacks the probabilistic explanation of the model and is not easy to be extended to handle the tasks with multiple source and target domains. Dai et al. [5] proposed a Co-clustering based approach for this problem. In this method, they identified the word clusters among the source and target domains, via which the class information and knowledge propagated from source domain to target domain.

Feature space mapping based methods are to map the original high-dimensional features into a low-dimensional feature space, under which the source and target domains comply with the same data distribution. Pan et al. [15] proposed a dimensionality reduction approach to find out this latent feature space, in which supervised learning algorithms can be applied to train classification models. Gu et al. [16] learnt the shared subspace among multiple domains for clustering and transductive transfer classification. In their problem formulation, all the domains have the same cluster centroid in the shared subspace. The label information can also be injected for classification tasks in this method. Xie et al. [17] tried to fill up those missing values of disjoint features to drive the marginal distributions of two domains closer, and then found the comparable substructures in the latent space where both marginal and conditional distribution are similar. In this laten space, given an unlabeled instances in the target domain the most similar labeled instances are retrieved for classification.

Weight based methods can be further grouped into two kinds, i.e. the instance weighting based and model weighting based methods. Instance weighting based approaches re-weight the instances in source domains according to the similarity measure on how they are close to the data in the target domain. Specifically, the weight of an instance is increased if it is close to the data in the target domain, otherwise the weight is decreased. Jiang et al. [18] proposed a general instance weighting framework, which has been validated to work well on NLP tasks. Dai et al. [7] extended boosting-style learning algorithm to cross-domain learning, in which the training instances with different distribution from the target domain are less weighted for data sampling, while the training instances with the similar distribution to the target domain are more weighted. On the other side model weighting based methods give different weights to the classification models in an ensemble. Gao et al. [2] proposed a dynamic model weighting method for each test example according to the similarity between the model and the local structure of the test example in the target domain.

Model combination based methods, considering the situation of multiple source domains, integrate the source-domain local models according to certain criterion. Ping at al [6] proposed the regularization framework which maximizes not only the posteriori in each source domain, but also the consensus degree of these models' prediction results on the target domain. Dredze at al [19] proposed a online model update method for each coming instance, which guarantee that after each iteration the combined model yields a correct prediction for the current instance with high probability while also making the smallest change from the existing models from the source domains.

The most related works are [20, 8]. The work of Zhai et al. [20] connects the variations of a topic under different contexts by leveraging the same background for this topic. Our work can also use this technique to explore possible improvements. In this sense, their work is orthogonal to ours. Xue et al. [8] proposed the model of topic-bridged PLSA for cross-domain text categorization, and the basic assumption of this work is that the source and target domains share the same topics. Specifically, they conduct two topic modelings over the source and target domains jointly, and induce the supervision of the labeled source domain data by the pair-wise constraints, similar to the must-link and cannot-link constraints used in semi-supervised clustering. Different from topic-bridged PLSA, our model explicitly explores the commonality (concept intension) and distinction (concept extension) of the topics across multiple domains rather than assume that these topics are exactly the same. Additionally, since our model has two latent variables for word concept and document class, it can naturally include the supervision from the source domain, rather than add a penalty of the pair-wise constraints to the original log-likelihood function.

6.2 Discussion on Generative vs. Discriminative Classifiers for Transfer Learning

Given the observed data x and their labels y, we can formulate the learning of a classifier as calculating the posterior distribution $p(y|x)$. A discriminative classifier models this distribution directly while a generative classifier models the joint probability $p(x, y)$, after which $p(y|x)$ is calculated via Bays rules. There is a widely-held belief in literatures that discriminative classifiers preferred to generative ones in practise. For example, Vapnik articulated in [21] that

> One should solve the classification problem directly and never solve a more general problem as an intermediate step such as modeling $p(x|y)$.

However, when learning and applying discriminative classifiers, we essentially assume that all the data instances are generated from the identical distribution. This assumption may not hold when data are from different sources. Ideally, the conditional probability $p(y|x)$ may be the same across different domains, however, the marginal probability $p(x)$ on each domain is prone to be different. The problem is that since the training of $p(y|x)$ based on the data in a source domain is biased towards the local marginal probability $p(x)$ it is difficult to achieve the ideal $p(y|x)$ by discriminative models even using the data from all the source domains. On the other hand, the generative classifiers, like CD-PLSA proposed here, provide us facilities to explicitly model the data distribution differences across domains. Thus, it may introduce extra values in prediction. Therefore, we argue that generative models may be suited for transfer learning.

7. CONCLUSIONS

In this paper, we investigated how to exploit the extension and intension of word and document concepts for cross-domain learning. The extension of word (document) concepts differs in various domain (*distinction*), but the intension of word (document) concepts is domain-independent (*commonality*). To this end, we proposed a CD-PLSA model to effectively capture the distinction and commonality across multiple domains for text classification, also developed an EM solution to it. Finally, the experimental results show that CD-PLSA significantly outperforms the baseline methods on the tasks with multiple source domains or multiple target domains, and it is more tolerant to distribution differences among the multiple domains.

8. ACKNOWLEDGEMENTS

The authors Fuzhen Zhuang, Qing He and Zhongzhi Shi are supported by the National Science Foundation of China (No. 60933004, 60975039), National Basic Research Priorities Programme (No.2007CB311004) and National Science and Technology Support Plan (No.2006BAC08B06).

9. REFERENCES

[1] W. Y. Dai, Y. Q. Chen, G. R. Xue, Q. Yang, and Y. Yu. Translated learning: Transfer learning across different feature spaces. In *Proceedings of the 22nd Neural Information Processing Systems (NIPS), Vancouver, British Columbia, Canada*, 2008.

[2] J. Gao, W. Fan, J. Jiang, and J. W. Han. Knowledge transfer via multiple model local structure mapping. In *Proceedings of the 14th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), Las Vegas, Nevada, USA*, pages 283–291, 2008.

[3] D. K. Xing, W. Y. Dai, G. R. Xue, and Y. Yu. Bridged refinement for transfer learning. In *Proceedings of the 11th Principles and Practice of Knowledge Discovery in Databases (PKDD), Warsaw, Poland*, pages 324–335, 2007.

[4] J. Gao, W. Fan, Y. Z. Sun, and J. W. Han. Heterogeneous source consensus learning via decision propagation and negotiation. In *Proceedings of the 15th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), Pairs, France*, 2009.

[5] W. Y. Dai, G. R. Xue, Q. Yang, and Y. Yu. Co-clustering based classification for out-of-domain documents. In *Proceedings of the 13th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), San Jose, California*, pages 210–219, 2007.

[6] P. Luo, F. Z. Zhuang, H. Xiong, Y. H. Xiong, and Q. He. Transfer learning from multiple source domains via consensus regularization. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM), Napa Valley, California, USA*, pages 103–112, 2008.

[7] W. Y. Dai, Q. Yang, G. R. Xue, and Y. Yu. Boosting for transfer learning. In *Proceedings of the 24th International Conference on Machine Learning (ICML)*, pages 193–200, 2007.

[8] G. R. Xue, W. Y. Dai, Q. Yang, and Y. Yu. Topic-bridged plsa for cross-domain text classification. In *Proc. of the 31st ACM Annual International Conference on Research and Development in Information Retrieval (SIGIR), Singapore*, pages 627–634, 2008.

[9] W. Y. Dai, O. Jin, G. R. Xue, Q. Yang, and Y. Yu. Eigen transfer: a unified framework for transfer learning. In *Proc. of the 26th Annual International Conference on Machine Learning (ICML), Montreal, Quebec, Canada*, pages 193–200, 2009.

[10] T. Hofmann. Probabilistic latent semantic analysis. In *Proceedings of 15th Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 289–296, 1999.

[11] Y. Jiho and S. J. Choi. Probabilistic matrix tri-factorization. In *Proc. of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1553–1556, 2009.

[12] David Hosmer and Stanley Lemeshow. *Applied Logistic Regression*. Wiley, New York, 2000.

[13] J. Jiang and C. X. Zhai. A two-stage approach to domain adaptation for statistical classifiers. In *Proceedings of the 16th ACM Conference on Information and Knowledge Management (CIKM)*, pages 401–410, 2007.

[14] F. Z. Zhuang, P. Luo, H. Xiong, Q. He, Y. H. Xiong, and Z. Z. Shi. Exploiting associations between word clusters and document classes for cross-domain text categorization. In *Proc. of the SIAM International Conference on Data Mining (SDM), Columbus, Ohio, USA*, pages 13–24, 2010.

[15] S. J. Pan, J. T. Kwok, and Q. Yang. Transfer learning via dimensionality reduction. In *Proceedings of the 23rd Conference on Artificial Intelligence (AAAI)*, pages 677–682, 2008.

[16] Q. Q. Gu and J. Zhou. Learning the shared subspace for multi-task clustering and transductive transfer classification. In *Proc. of the International Conference on Data Mining (ICDM), Miami, Florida, USA*, 2009.

[17] S. H. Xie, W. Fan, J. Peng, O. Verscheure, and J. T. Ren. Latent space domain transfer between high dimensional overlapping distributions. In *Proc. of ACM Conference on World Wide Web (WWW), Madrid, Spain*, pages 91–100, 2009.

[18] J. Jiang and C. X. Zhai. Instance weighting for domain adaptation in nlp. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 264–271, 2007.

[19] M. Dredze, A. Kulesza, and K. Crammer. Multi-domain learning by confidence-weighted parameter combination. *Journal of Machine Learning*, 2009.

[20] C. X. Zhai, A. Velivelli, and B. Yu. A cross-collection mixture modle for comparative text mining. In *Proceedings of the 10th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), Seattle, Washington, USA*, pages 743–748, 2004.

[21] V. N. Vapnik. *Statictic Learning Theory*. New York: Wiely-Interscience, 1998.

APPENDIX

First, we now consider the log joint probability $\log p(\boldsymbol{Z}, \boldsymbol{X}|\theta)$ and the posterior probability of the latent factors $p(\boldsymbol{Z}|\boldsymbol{X}; \theta^{\text{old}})$ separately.

According to Figure 2 and the d-separation criterion, we have

$$\log p(\boldsymbol{Z}, \boldsymbol{X}|\theta) = \log \prod_n p(\boldsymbol{Z}_n, \boldsymbol{X}_n|\theta) = \sum_n \log p(\boldsymbol{Z}_n, \boldsymbol{X}_n|\theta), \tag{21}$$

where $\boldsymbol{X}_n, \boldsymbol{Z}_n$ are the n-th entries of $\boldsymbol{X}$ and $\boldsymbol{Z}$ respectively.

Similarly, we have

$$p(\boldsymbol{Z}|\boldsymbol{X}; \theta) = \prod_m p(\boldsymbol{Z}_m|\boldsymbol{X}; \theta) = \prod_m p(\boldsymbol{Z}_m|\boldsymbol{X}_m; \theta) \tag{22}$$

Then $\mathcal{L}$ become (using (21) and (22)):

$$\begin{aligned}
\mathcal{L} &= \sum_{\boldsymbol{Z}} p(\boldsymbol{Z}|\boldsymbol{X}; \theta^{\text{old}}) \log p(\boldsymbol{Z}, \boldsymbol{X}|\theta) \\
&= \sum_{\boldsymbol{Z}} \prod_m p(\boldsymbol{Z}_m|\boldsymbol{X}_m; \theta^{\text{old}}) \sum_n \log p(\boldsymbol{Z}_n, \boldsymbol{X}_n|\theta) \\
&= \sum_n \sum_{\boldsymbol{Z}} \prod_m p(\boldsymbol{Z}_m|\boldsymbol{X}_m; \theta^{\text{old}}) \log p(\boldsymbol{Z}_n, \boldsymbol{X}_n|\theta) \\
&= \sum_n \sum_{\boldsymbol{Z}_n} \sum_{\boldsymbol{Z}_{-n}} \prod_{m \neq n} p(\boldsymbol{Z}_m|\boldsymbol{X}_m; \theta^{\text{old}}) \\
&\quad \cdot p(\boldsymbol{Z}_n|\boldsymbol{X}_n; \theta^{\text{old}}) \log p(\boldsymbol{Z}_n, \boldsymbol{X}_n|\theta) \\
&= \sum_n \sum_{\boldsymbol{Z}_n} p(\boldsymbol{Z}_n|\boldsymbol{X}_n; \theta^{\text{old}}) \log p(\boldsymbol{Z}_n, \boldsymbol{X}_n|\theta) \\
&\quad \cdot \sum_{\boldsymbol{Z}_{-n}} \prod_{m \neq n} p(\boldsymbol{Z}_m|\boldsymbol{X}_m; \theta^{\text{old}}) \\
&= \sum_n \sum_{\boldsymbol{Z}_n} p(\boldsymbol{Z}_n|\boldsymbol{X}_n; \theta^{\text{old}}) \log p(\boldsymbol{Z}_n, \boldsymbol{X}_n|\theta).
\end{aligned} \tag{23}$$

Now we write the observed data $\boldsymbol{X}_n$ in detail as (w, d, c), each component of $\boldsymbol{Z}_n$ as (y, z). Then, we have

$$\begin{aligned}
\mathcal{L} &= \sum_{w,d,c} O_{w,d,c} \sum_{y,z} p(y, z|w, d, c; \theta^{\text{old}}) \\
&\quad \cdot \log p(y, z, w, d, c|\theta) \\
&= \sum_{y,z,w,d,c} O_{w,d,c} p(y, z|w, d, c; \theta^{\text{old}}) \\
&\quad \cdot \log [p(y, z) p(w|y, c) p(d|z, c) p(c)],
\end{aligned} \tag{24}$$

where $O_{w,d,c}$ is the co-occurrence number of w, d, c.