Concept Learning for Cross-Domain Text Classification: A General Probabilistic Framework

Fuzhen Zhuang[1], **Ping Luo**[2], **Peifeng Yin**[3], **Qing He**[1], **Zhongzhi Shi**[1]
[1]Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences. {zhuangfz,heq,shizz}@ics.ict.ac.cn
[2]Hewlett Packard Labs, China. ping.luo@hp.com
[3]Pennsylvania State University. pzy102@cse.psu.edu

Abstract

Cross-domain learning targets at leveraging the knowledge from source domains to train accurate models for the test data from target domains with different but related data distributions. To tackle the challenge of data distribution difference in terms of raw features, previous works proposed to mine high-level *concepts* (e.g., word clusters) across data domains, which shows to be more appropriate for classification. However, all these works assume that the same set of concepts are shared in the source and target domains in spite that some *distinct concepts* may exist only in one of the data domains. Thus, we need a general framework, which can incorporate both shared and distinct concepts, for cross-domain classification. To this end, we develop a probabilistic model, by which both the shared and distinct concepts can be learned by the EM process which optimizes the data likelihood. To validate the effectiveness of this model we intentionally construct the classification tasks where the distinct concepts exist in the data domains. The systematic experiments demonstrate the superiority of our model over all compared baselines, especially on those much more challenging tasks.

1 Introduction

Traditional machine learning algorithms often fail to obtain satisfactory performance when the training and test data are drawn from different but related data distributions. See the task example as follows. To build an enterprize news portal we need to classify the news about a certain company into some predefined categories, such as "product announcement", "financial analysis", "employee laid-off" 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. Thus, cross-domain learning (referred as domain adaptation or transfer learning), focusing on adapting the models trained from source domains to target domains, attracts many research interests recently [Pan and Yang, 2010; Dai *et al.*, 2008; Gao *et al.*, 2008; Dai *et al.*, 2007a; Luo *et al.*, 2008; Xue *et al.*, 2008; Dai *et al.*, 2009; Zhuang *et al.*, 2010b; 2010a; Wang *et al.*, 2011; Long *et al.*, 2012]. Some of these previous works have showed that the high-level *word concepts* are more appropriate for across-domain text classification instead of the raw word features. Specifically, they consider two sides of a concept, namely concept *extension* and *intension*.

Definition 1 (Extension of Word Concept) *The extension of a word concept z is represented by a multinomial distribution $p(w|z)$ over words.*

It is actually the degree of applicability of each word w for that concept z. That is to say, when $p(w|z)$ is large, w is a word to the word concept z.

Definition 2 (Intension of Word Concept) *The intension of a word concept z is expressed by its association with each document class y, denoted by the conditional probability $p(z|y)$ in this study.*

It indicates that when $p(z|y)$ is large, the concept z is strongly related to the document class y.

Zhuang et al. [Zhuang *et al.*, 2010b; 2010a] showed that the extension of a concept is often domain-dependent. See the concept of "product" in the task of enterprize news classification. If this concept is in the domain of HP (which makes printers), the values of $p(\text{“}printer\text{”}|\text{“}products\text{”}, HP)$ and $p(\text{“}LaserJet\text{”}|\text{“}products\text{”}, HP)$ are large. However, these two values $p(\text{“}printer\text{”}|\text{“}products\text{”}, IBM)$ and $p(\text{“}LaserJet\text{”}|\text{“}products\text{”}, IBM)$ will be very small in the domain of IBM, since IBM seldom sells printers. Additionally, their work observed that the intension of a concept, namely $p(z|y)$, is usually stable across domains. In other words, wherever a word concept exists, it has the same implication to the class of the document. For example, 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 "employee laid-off". Based on these observations we give the definition of *homogeneous concepts* as follows.

Definition 3 (Homogeneous Concepts) *A concept z^b is homogeneous if it has the **same intension** but **different extensions** across the data domains.*

In other words, since the intension of a homogeneous concept z^b remains the same across data domains we have

$$p(z^b|y, r) = p(z^b|y, r'),$$

where r, r' are two of any data domains. Thus, to represent the intension of a homogeneous concept we do not need the variable r of data domain in the conditional probability. Thus, it can be represented as $p(z^b|y)$. However, since the extension of a homogeneous concept changes in different data domains we have

$$p(w|z^b, r) \neq p(w|z^b, r').$$

Thus, the extension of a homogeneous concept can only be represented as $p(w|z^b, r)$, where the variable r of data domain is included in the condition.

Besides the homogeneous concepts, the data domains may also share some concepts with the same intension and extension. For example, the enterprize news about "financial analysis" may talk about the concept of "accounting terminologies". Different companies may use the same set of standard terminologies. Thus, the intension of this concept remains the same even in different domains. Thus, we can give the definition of *identical concepts* as follows.

Definition 4 (Identical Concepts) *A concept z^a is identical if it has the **same intension** and the **same extension** across the data domains.*

Similarly, for an identical concept z^a we have

$$p(z^a|y, r) = p(z^a|y, r'), p(w|z^a, r) = p(w|z^a, r'),$$

where r, r' are two of any data domains. Therefore, the intension and extension of an identical concept can be represented as $p(z^a|y), p(w|z^a)$ respectively, where the variable r of data domain can be omitted.

It is clear that both homogeneous and identical concepts are shared among all the data domains. In this study we further argue that some concepts may only exist in one of the data domains. See the task of enterprize news classification again. Assume that one of the companies is in the process of stock market launch. Thus, the news about this company may mention the concept about the "IPO process". Since all the other companies have been listed for a long time, the concept about "IPO process" may only exist in the news about this company. With this observation we can give the definition of *distinct concepts* as follows.

Definition 5 (Distinct Concepts) *A concept z^c is distinct if it only exists in one of the data domains.*

If a distinct concept z^c exists only in the data domain r, its intension and extension can be represented as $p(z^c|y, r), p(w|z^c, r)$ respectively. However, for any other domain r' the values of $p(z^c|y, r'), p(w|z^c, r')$ are *invalid*.

In summary we observe three kinds of concepts in the data domains for transfer learning, as shown in Figure 1. The homogeneous and identical concepts are shared among all the data domains, while the distinct concepts may only exist in one of them. Thus, we need a general model to capture all these concepts in all the data domains simultaneously. Hopefully, it may lead to better classification performance with the increase of the model expression ability. Along this line, we propose a *Homogeneous-Identical-Distinct-Concept* (HIDC for short) model based on the probabilistic methods. Then, an EM algorithm is developed to solve the likelihood maximum problem. Finally, we conduct systemic experiments to show the effectiveness of the proposed model as well as its superiority over the compared baselines.

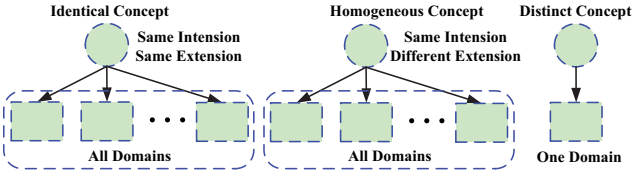


Figure 1: The Identical, Homogeneous, and Distinct Concepts

2 Preliminary Knowledge and Problem Formulation

In this section, we first briefly review Probabilistic Latent Semantic Analysis (PLSA) [Hofmann, 1999], and then introduce an extension of PLSA, Dual-PLSA [Jiho and Choi, 2009]. Finally, we formulate the optimization problem for cross-domain text classification which incorporates all the three kinds of concepts.

2.1 Preliminary on PLSA and D-PLSA

Probabilistic Latent Semantic Analysis [Hofmann, 1999] 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 z) as in Eq.(1).

$$p(w, d) = \sum_z p(w, d, z) = \sum_z p(w|z)p(d|z)p(z). \quad (1)$$

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

(a) PLSA (b) D-PLSA

Figure 2: The Graphical Models of PLSA and D-PLSA

In the PLSA model, the documents and words share the same latent variable z. However, documents and words usually exhibit different organizations and structures. Specifically, they may have different kinds of latent topics, denoted by z for word concept and y for document cluster. 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) [Jiho and Choi, 2009] in this paper.

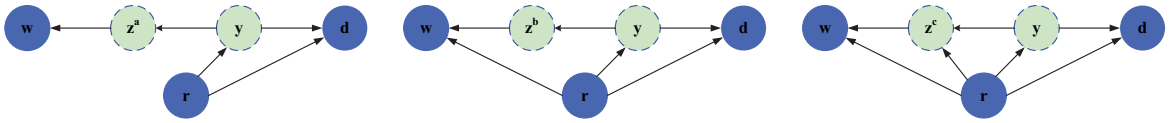
Given the word-document co-occurrence $\boldsymbol{O}$, we can similarly arise a mixture model like Eq.(1),

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

And the parameters of $p(w|z), p(d|y), p(z|y), p(y)$ over all w, d, z, y can also be obtained by the EM solution.

2.2 The Graphical Models for Identical, Homogeneous, and Distinct Concepts

Motivated by D-PLSA, we propose three probabilistic generative models in Figure 3, which correspond to the three kinds of concepts in Definitions 3 through 5. In these sub-figures,



(a) Considering the Identical Concepts (b) Considering the Homogeneous Concepts (c) Considering the Distinct Concepts

Figure 3: The Graphical Models

w and d refer to any word and any document respectively, z and y refer to any word concept and any document cluster respectively, and r refers to any data domain. Additionally, the word concept z can be further divided into three kinds, namely z^a, z^b, and z^c, representing identical concepts, homogeneous concepts, and distinct concepts respectively.

First, Figure 3(a) considers the identical concepts, where both the concept extension $p(w|z^a)$ and intension $p(z^a|y)$ are irrelevant to the data domain r. It indicates that no matter where z^a comes from it has the same extension and intension. Second, Figure 3(b) considers the homogeneous concepts. You can see that in Figure 3(b) the concept extension $p(w|z^b, r)$ is dependent on r while the intension $p(z^b|y)$ is irrelevant to r. It indicates that any homogeneous concept z^b has the stable intension but different extensions across the domains. Finally, Figure 3(c) considers the distinct concepts. Here, since any distinct concept z^c exists only in r, its extension $p(w|z^c, r)$ and intension $p(z^c|y, r)$ are both relevant to the data domain r.

Given the graphical models in Figure 3, the joint probability over all the variables is

$$\begin{aligned} p(w,d,r) &= \sum_{z,y} p(z,y,w,d,r) = \sum_{z^a,y} p(z^a,y,w,d,r) \\ &+ \sum_{z^b,y} p(z^b,y,w,d,r) + \sum_{z^c,y} p(z^c,y,w,d,r). \end{aligned} \tag{3}$$

Here, the concept z is expanded by z^a, z^b, z^c. Additionally, based on the graphical models in Figures 3(a), 3(b) and 3(c), we have

$$p(z^a,y,w,d,r) = p(w|z^a)p(z^a|y)p(d|y,r)p(y|r)p(r), \tag{4}$$

$$p(z^b,y,w,d,r) = p(w|z^b,r)p(z^b|y)p(d|y,r)p(y|r)p(r), \tag{5}$$

$$p(z^c,y,w,d,r) = p(w|z^c,r)p(z^c|y,r)p(d|y,r)p(y|r)p(r). \tag{6}$$

2.3 Problem Formulation

Suppose we have $s+t$ data domains, denoted as $\boldsymbol{X} = (\boldsymbol{X}_1, \cdots, \boldsymbol{X}_s, \boldsymbol{X}_{s+1}, \cdots, \boldsymbol{X}_{s+t})$. Without loss of generality, we assume the first s domains are source domains with label information, i.e., $\boldsymbol{X}_r = \{x_i^{(r)}, y_i^{(r)}\}|_{i=1}^{n_r}$ $(1 \le r \le s)$, and the left t domains are target domains without any label information, i.e., $\boldsymbol{X}_r = \{x_i^{(r)}\}|_{i=1}^{n_r}$ $(s+1 \le r \le s+t)$, n_r is the number of documents in data domain $\boldsymbol{X}_r$.

We try to maximize the log likelihood in Eq.(7) as follows,

$$\log p(\boldsymbol{X};\theta) = \log \sum_{\boldsymbol{Z}} p(\boldsymbol{Z},\boldsymbol{X};\theta), \tag{7}$$

where $\boldsymbol{Z}$ includes all the latent variables, and θ represents all the model parameters.

In the following section, we develop an EM algorithm to solve the optimization problem in Eq.(7).

3 Solution to HIDC

An Expectation-Maximization (EM) algorithm is to maximize the lower bound (via Jensen's inequality) $\mathcal{L}_0$ of Eq.(7):

$$\mathcal{L}_0 = \sum_{\boldsymbol{Z}} q(\boldsymbol{Z}) \log \{\frac{p(\boldsymbol{Z},\boldsymbol{X};\theta)}{q(\boldsymbol{Z})}\}, \tag{8}$$

where $q(\boldsymbol{Z})$ could be arbitrary probability distribution over all latent variables. For computation convenience, we set $q(\boldsymbol{Z}) = p(\boldsymbol{Z}|\boldsymbol{X};\theta^{\text{old}})$, and omit the constant. Then the objective function $\mathcal{L}_0$ becomes $\mathcal{L}$,

$$\begin{aligned} \mathcal{L} &= \sum_{\boldsymbol{Z}} p(\boldsymbol{Z}|\boldsymbol{X};\theta^{\text{old}}) \log p(\boldsymbol{Z},\boldsymbol{X};\theta) \\ &= \sum_{\boldsymbol{Z}^a} p(\boldsymbol{Z}^a|\boldsymbol{X};\theta^{\text{old}}) \log p(\boldsymbol{Z}^a,\boldsymbol{X};\theta) \\ &+ \sum_{\boldsymbol{Z}^b} p(\boldsymbol{Z}^b|\boldsymbol{X};\theta^{\text{old}}) \log p(\boldsymbol{Z}^b,\boldsymbol{X};\theta) \\ &+ \sum_{\boldsymbol{Z}^c} p(\boldsymbol{Z}^c|\boldsymbol{X};\theta^{\text{old}}) \log p(\boldsymbol{Z}^c,\boldsymbol{X};\theta). \end{aligned} \tag{9}$$

Here, we denote z^a, y as $\boldsymbol{Z}^a$, z^b, y as $\boldsymbol{Z}^b$, z^c, y as $\boldsymbol{Z}^c$.

3.1 E-step

Suppose $\boldsymbol{X}_r$ includes the variables w, d, r, and $\boldsymbol{O}_r$ corresponds to word-document co-occurrence of data $\boldsymbol{X}_r$ $(1 \le r \le s+t)$, we can obtain Eq.(10) by substituting all variables into Eq.(9),

$$\begin{aligned} \mathcal{L} = &\sum_{w,d,z^a,y,r} \boldsymbol{O}_{w,d,r} p(z^a,y|w,d,r;\theta^{\text{old}}) \log p(z^a,y,w,d,r;\theta) \\ + &\sum_{w,d,z^b,y,r} \boldsymbol{O}_{w,d,r} p(z^b,y|w,d,r;\theta^{\text{old}}) \log p(z^b,y,w,d,r;\theta) \\ + &\sum_{w,d,z^c,y,r} \boldsymbol{O}_{w,d,r} p(z^c,y|w,d,r;\theta^{\text{old}}) \log p(z^c,y,w,d,r;\theta), \end{aligned} \tag{10}$$

where $O_{w,d,r}$ represents the co-occurrence frequency of the triple (w,d,r). Note that $p(z^a,y,w,d,r)$, $p(z^b,y,w,d,r)$ and $p(z^c,y,w,d,r)$ can be respectively calculated by Eqs.(4), (5) and (6). Therefore, the iterative formulas in E-step are as follows,

$$\hat{p}(z^a,y|w,d,r) = \frac{p(w|z^a)p(z^a|y)p(d|y,r)p(y|r)p(r)}{\sum_{z^a,y} p(w|z^a)p(z^a|y)p(d|y,r)p(y|r)p(r)}, \tag{11}$$

$$\hat{p}(z^b,y|w,d,r) = \frac{p(w|z^b,r)p(z^b|y)p(d|y,r)p(y|r)p(r)}{\sum_{z^b,y} p(w|z^b,r)p(z^b|y)p(d|y,r)p(y|r)p(r)}, \tag{12}$$

$$\hat{p}(z^c,y|w,d,r) = \frac{p(w|z^c,r)p(z^c|y,r)p(d|y,r)p(y|r)p(r)}{\sum_{z^c,y} p(w|z^c,r)p(z^c|y,r)p(d|y,r)p(y|r)p(r)}. \tag{13}$$

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

Now we maximize $\mathcal{L}$ with its parameters by Lagrangian Multiplier method. Take the derivation of $p(d|y,r)$ as an example, and extract the terms containing $p(d|y,r)$. Then, we have

$$\begin{aligned}\mathcal{L}_{[p(d|y,r)]} = &\sum_{w,d,z^a,y,r} O_{w,d,r} p(z^a, y|w,d,r;\theta^{\text{old}}) \cdot \log p(d|y,r) \\ &+ \sum_{w,d,z^b,y,r} O_{w,d,r} p(z^b, y|w,d,r;\theta^{\text{old}}) \cdot \log p(d|y,r) \\ &+ \sum_{w,d,z^c,y,r} O_{w,d,r} p(z^c, y|w,d,r;\theta^{\text{old}}) \cdot \log p(d|y,r).\end{aligned} \tag{14}$$

Applying the constraint $\sum_d p(d|y,r) = 1$ into the following equation:

$$\frac{\partial \left[\mathcal{L}_{[p(d|y,r)]} + \lambda(1 - \sum_d p(d|y,r))\right]}{\partial p(d|y,r)} = 0, \tag{15}$$

then

$$\begin{aligned}&\lambda \cdot p(d|y,r) = \sum_{w,z^a} O_{w,d,r}\ p(z^a, y|w,d,r;\theta^{\text{old}}) + \\ &\sum_{w,z^b} O_{w,d,r}\ p(z^b, y|w,d,r;\theta^{\text{old}}) + \sum_{w,z^c} O_{w,d,r}\ p(z^c, y|w,d,r;\theta^{\text{old}}).\end{aligned} \tag{16}$$

Considering the constraint $\sum_d p(d|y,r) = 1$,

$$\begin{aligned}&\Rightarrow \lambda = \sum_{w,d,z^a} O_{w,d,r}\ p(z^a, y|w,d,r;\theta^{\text{old}}) + \\ &\sum_{w,d,z^b} O_{w,d,r}\ p(z^b, y|w,d,r;\theta^{\text{old}}) + \sum_{w,d,z^c} O_{w,d,r}\ p(z^c, y|w,d,r;\theta^{\text{old}}).\end{aligned} \tag{17}$$

Finally, the update formula of $p(d|y,r)$ can be obtained,

$$\begin{aligned}\hat{p}(d|y,r) \propto &\sum_{w,z^a} O_{w,d,r} p(z^a, y|w,d,r) \\ &+ \sum_{w,z^b} O_{w,d,r} p(z^b, y|w,d,r) + \sum_{w,z^c} O_{w,d,r} p(z^c, y|w,d,r),\end{aligned} \tag{18}$$

Similarly,

$$\hat{p}(w|z^a) = \frac{\sum_{d,y,r} O_{w,d,r} p(z^a, y|w,d,r)}{\sum_{w,d,y,r} O_{w,d,r} p(z^a, y|w,d,r)}, \tag{19}$$

$$\hat{p}(w|z^b, r) = \frac{\sum_{d,y} O_{w,d,r} p(z^b, y|w,d,r)}{\sum_{w,d,y} O_{w,d,r} p(z^b, y|w,d,r)}, \tag{20}$$

$$\hat{p}(w|z^c, r) = \frac{\sum_{d,y} O_{w,d,r} p(z^c, y|w,d,r)}{\sum_{w,d,y} O_{w,d,r} p(z^c, y|w,d,r)}, \tag{21}$$

$$\hat{p}(z^a|y) = \frac{\sum_{w,d,r} O_{w,d,r} p(z^a, y|w,d,r)}{\sum_{w,d,z^a,r} O_{w,d,r} p(z^a, y|w,d,r)}, \tag{22}$$

$$\hat{p}(z^b|y) = \frac{\sum_{w,d,r} O_{w,d,r} p(z^b, y|w,d,r)}{\sum_{w,d,z^b,r} O_{w,d,r} p(z^b, y|w,d,r)}, \tag{23}$$

$$\hat{p}(z^c|y,r) = \frac{\sum_{w,d} O_{w,d,r} p(z^c, y|w,d,r)}{\sum_{w,d,z^c} O_{w,d,r} p(z^c, y|w,d,r)}, \tag{24}$$

$$\begin{aligned}\hat{p}(y|r) \propto &\sum_{w,d,z^a} O_{w,d,r} p(z^a, y|w,d,r) \\ &+ \sum_{w,d,z^b} O_{w,d,r} p(z^b, y|w,d,r) + \sum_{w,d,z^c} O_{w,d,r} p(z^c, y|w,d,r),\end{aligned} \tag{25}$$

$$\begin{aligned}\hat{p}(r) \propto &\sum_{w,d,z^a,y} O_{w,d,r} p(z^a, y|w,d,r) \\ &+ \sum_{w,d,z^b,y} O_{w,d,r} p(z^b, y|w,d,r) + \sum_{w,d,z^c,y} O_{w,d,r} p(z^c, y|w,d,r).\end{aligned} \tag{26}$$

3.3 HIDC to Cross-domain Classification

In this subsection, we introduce how to leverage the proposed EM algorithm for cross-domain text classification. There are 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.

Algorithm 1 HIDC for Cross-domain Text Classification

Input: Given $(s+t)$ data domains $\boldsymbol{X}_1, \cdots, \boldsymbol{X}_s, \boldsymbol{X}_{s+1}, \cdots, \boldsymbol{X}_{s+t}$, where the first s domains are source domains with label information, while the left are target domains. T, the number of iterations. k_1, k_2, k_3, the number of identical concepts, homogeneous concepts and distinct concepts. (In this study we simply assume each domain has the same number of distinct concepts.)

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

1. Initialization. The detailed initialization of $p^{(0)}(w|z^a)$, $p^{(0)}(w|z^b, r)$, $p^{(0)}(w|z^c, r)$ can be referred in Section 4.2. The initialization of $p^{(0)}(d|y,r)$ is detailed in Section 3.3. $p^{(0)}(z^a|y)$, $p^{(0)}(z^b|y)$, $p^{(0)}(z^c|y,r)$, $p(y|r)$ and $p(r)$ are set randomly.
2. $k := 1$.
3. for $r := 1 \rightarrow s+t$
   Update $p^{(k)}(z^a, y|w,d,r)$, $p^{(k)}(z^b, y|w,d,r)$ and $p^{(k)}(z^c, y|w,d,r)$ according to Eqs.(11), (12) and (13) in **E-step**;
   Update $p^{(k)}(w|z^b, r)$ and $p^{(k)}(w|z^c, r)$ according to Eqs.(20) and (21) in **M-step**;
   Update $p^{(k)}(z^c|y,r)$, $p^{(k)}(y|r)$ and $p(r)$ according to Eqs.(24), (25) and (26) in **M-step**;
4. end.
5. Update $p^{(k)}(w|z^a)$, $p^{(k)}(z^a|y)$ and $p^{(k)}(z^b|y)$ according to Eqs.(19), (22) and (23) in **M-step**;
6. for $r := s+1 \rightarrow s+t$
   Update $p^{(k)}(d|y,r)$ according to Eq.(18) in **M-step**;
7. end.
8. $k := k+1$, if $k < T$, turn to Step 3.
9. The class label of any document d in a target domain is predicted by Eqs.(27) and (28).

For the first task we inject the supervising information (the class label of the instances in the source domains) into the probability $p(d|y,r)$ ($1 \le r \le s$). Specifically, let $\boldsymbol{L}^r \in \{0,1\}^{n_r \times m}$ be the true label information of the r-th domain, where n_r is the number of instances in it, m is the number of document classes. If instance d belongs to document class y_0, then $L^r_{d,y_0} = 1$, otherwise $L^r_{d,y} = 0$ ($y \neq y_0$). We normalize $\boldsymbol{L}^r$ to satisfy the probability condition so that the sum of the entries in each column equals to 1, $N^r_{d,y} = \frac{L^r_{d,y}}{\sum_d L^r_{d,y}}$. Then $p(d|y,r)$ is initialized as $N^r_{d,y}$. Note that since this initial value is from the true class label we do not change the value of $p(d|y,r)$ (for $1 \le r \le s$) during the iterative process.

For the unlabeled target domains, $p(d|y,r)$ ($s+1 \leq r \leq s+t$) can be initialized similarly. This time the label information $\boldsymbol{L}^r$ 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 $p(d|y,r)$ (for $s+1 \leq r \leq s+t$) during the iterative process.

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

$$\begin{aligned} p(y|d,r) &= \frac{p(y,d,r)}{p(d,r)} \propto p(y,d,r) = p(d|y,r)p(y,r) \\ &= p(d|y,r)p(y|r)p(r). \end{aligned} \tag{27}$$

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

$$\arg\max_y p(y|d,r). \tag{28}$$

The detailed procedure of HIDC for cross-domain text classification is depicted in Algorithm 1.

4 Experimental Evaluation

In this study, we focus on evaluating our model on binary classification scenarios. Note that our model can naturally handle multi-class classification tasks.

4.1 Data Preparation

20Newsgroups[1] is one of the mostly used data sets for evaluating transfer learning algorithms [Pan and Yang, 2010; Gao *et al.*, 2008; Dai *et al.*, 2007b; Zhuang *et al.*, 2010a]. This corpus has approximately 20,000 newsgroup documents, which are evenly divided into 20 subcategories. Some similar subcategories are grouped into a top category, e.g., the four subcategories *sci.crypt*, *sci.electronics*, *sci.med* and *sci.space* belong to the top category *sci*. We use four top categories *comp*, *rec*, *sci* and *talk* to construct classification tasks, and each top category contains four subcategories.

To validate the effectiveness of HIDC, we construct the transfer learning problems as follows. Firstly, we choose two top categories *rec* and *sci*, one as positive class and the other one as negative one. To form a source domain, we randomly select one subcategory from *rec* and another subcategory from *sci*. The target domain is similarly constructed. Thus in this way we can obtain 144 ($P_4^2 \times P_4^2$) tasks. In these tasks since the documents in the source and target domains are drawn from the same two top categories, and they may share lots of homogeneous and identical concepts, but have less distinct concepts. Thus, we call this kind of tasks as tasks with *less* distinct concepts.

Secondly, to intentionally construct the classification scenario when the distinct concepts may exist, we replace one subcategory for the target domain with another subcategory from the top category *comp* or *talk*. For example, the source domain has the documents from the top categories of *rec* and *sci*, while the target domain has the ones from *rec* and *talk*. Then, totally 384 ($P_4^2 \times P_4^1 \times 8$) problems can be constructed. In these tasks since the documents in the target domain may come from the top category which is not included in the source domain, there may exist lots of distinct concepts in them. Thus, these *new* transfer learning tasks might be much more challenging. To avoid negative transfer, among all the 384 tasks we only consider the 334 tasks on which the supervised learning model Logistic Regression (LR) [Hosmer and Lemeshow, 2000] outputs the accuracy higher than 50%. We call this kind of tasks as tasks with *more* distinct concepts.

In summary, we have 144 *traditional* transfer learning tasks and 334 *new* ones from the *20Newsgroups* data set.

Sentiment Data. We use the Multi-Domain sentiment data set[2] [Blitzer *et al.*, 2007], which contains product reviews taken from Amazon.com for many product types (domains) to further demonstrate the effectiveness of HIDC. This data set contains the positive and negative reviews from four domains, i.e., books, dvd, electronics and kitchen.

Here, to show that HIDC can naturally deal with the classification scenarios which have multiple source and target domains, we construct the tasks with two source domains and two target domains from the Sentiment Data. Specifically, we randomly select two domains as source domains, and the rest two as target domains. Thus we have 6 classification problems.

4.2 Experimental Setup

Compared algorithms: We compare our model HIDC with some state-of-the-art baselines, including 1) the supervised algorithms: Logistic Regression (LR) [Hosmer and Lemeshow, 2000], Support Vector Machine (SVM) [Boser *et al.*, 1992]; 2) the semi-supervised algorithm: Transductive Support Vector Machine (TSVM) [Joachims, 1999]; 3) the cross-domain methods: Co-clustering based Classification (CoCC) [Dai *et al.*, 2007a], CD-PLSA [Zhuang *et al.*, 2010a] and Dual Transfer Learning (DTL) [Long *et al.*, 2012].

Parameter setting: In Algorithm 1, we set $k_1 = 20$, $k_2 = 20$, $k_3 = 10$ and $T = 100$ (Additional experiments show that HIDC is not sensitive to the parameter setting when they are sampled from some predefined bounds, and converges very fast. Due to the space limitation, we don't detail them here.). The parameters $p(w|z^a)$, $p(w|z^b,r)$ and $p(w|z^c,r)$ ($1 \leq r \leq s+t$) are initialized as follows. We combine all the data from source and target domains, and conduct the PLSA implemented by Matlab[3]. We set the number of topics as (k_1+k_2), and obtain the word clustering information $W \in \mathbb{R}_+^{m\times(k_1+k_2)}$ (m is the number of word features). W is divided into two parts $W=[W^1,W^2]$ ($W^1 \in \mathbb{R}_+^{m\times k_1}$, $W^2 \in \mathbb{R}_+^{m\times k_2}$), then $p(w|z^a)$ is initialized as W^1 and $p(w|z^b,r)$ is assigned as W^2. Finally, $p(w|z^c,r)$ is randomly initialized, while ensuring $\sum_w p(w|z^c,r)=1$. Due to the random initialization, we run ten times for each problem and report the average performance. As for the baseline methods, LR is implemented by Matlab[4], SVM and TSVM are given by SVM^{light}[5]. The parameters of CoCC, CD-PLSA and DTL are set as the default ones in their orig-

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

[2] http://www.cs.jhu.edu/ mdredze/datasets/sentiment/.

[3] http://www.kyb.tuebingen.mpg.de/bs/people/pgehler/code/.

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

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

Table 1: Average Performances (%) on The New and Traditional Transfer Learning Tasks

		LR	SVM	TSVM	CoCC	DTL	CD-PLSA	HIDC
Tasks With More Distinct Concepts	*Lower than 55%*	52.45	51.81	74.32	69.66	75.34	84.77	**90.12**
	Higher than 55%	70.03	68.62	86.68	91.76	94.44	95.36	**96.50**
	Total	66.60	65.34	84.27	87.45	90.71	93.30	**95.25**
Tasks With Less Distinct Concepts	*Lower than 55%*	52.47	51.82	69.88	64.09	68.62	84.35	**93.13**
	Higher than 55%	68.33	66.80	85.52	91.84	94.18	96.00	**96.12**
	Total	65.57	64.20	82.81	87.02	89.75	93.97	**95.60**

inal papers, since their authors have demonstrated they can perform well on the *20Newsgroups* data set. We use the classification accuracy as the evaluation metric.

4.3 Experimental Results

Results on the 20NewsGroup Data

For the 334 tasks with more distinct concepts we further divide them into two parts, i.e., 65 tasks whose accuracies from LR are lower than 55%, and the other 269 ones higher than 55%. The average accuracies on these tasks for all the compared algorithms are listed in Table 1.

When the accuracies from LR are lower than 55%, we can find that 1) these problems are very difficult, since both of the supervised machine learning algorithms LR and SVM perform only slightly better than random classification. 2) HIDC achieves the outstanding results, which is significantly better than all the other algorithms. Especially, HIDC can obtain 37.67% average accuracy improvement compared with LR. 3) HIDC performs better than TSVM, since TSVM still assumes the source and target domains are drawn from the same data distributions. 4) CD-PLSA is better than DTL, and DTL is better than CoCC. In summary these results show that modeling the distinct concepts in HIDC helps to significantly improve the classification performance.

When the accuracies from LR are higher than 55%, we can still see that all the four cross-domain algorithms perform better than the traditional supervised algorithms, and HIDC still outperforms all other ones. We can also see that the performance improvement from HIDC under this situation is less than that for the tasks whose LR accuracies are less than 55%.

Similarly, we can also divide the 144 tasks with less distinct concepts into two parts, i.e., with the LR accuracy higher or lower than 55%. As shown in Table 1, again HIDC beats all the baselines.

Table 2: The Performance Comparison (%) on Sentiment Data Set

	LR		DTL		CD-PLSA		HIDC	
	Tar 1	Tar 2	Tar 1	Tar 2	Tar 1	Tar 2	Tar 1	Tar 2
Task 1	74.35	75.30	76.94	77.42	78.14	81.31	82.81	82.76
Task 2	71.70	74.35	75.98	77.42	69.91	75.79	75.13	75.01
Task 3	70.80	84.25	73.54	80.11	74.89	82.92	83.82	83.85
Task 4	71.7	82.6	74.36	79.38	70.88	79.56	81.78	81.89
Task 5	74.35	82.6	77.29	79.32	70.69	79.17	81.92	81.87
Task 6	69.95	84.25	75.85	78.47	67.61	81.97	83.57	83.6
Average	72.14	80.56	75.66	78.69	72.02	80.12	**81.51**	**81.50**

Results on Sentiment Data

Note that the tasks on Sentiment Data have the two source domains and two target domains. For the supervised algorithm LR [Hosmer and Lemeshow, 2000], we record the ensemble performance of the classifiers from the two source domains. Also, we compare HIDC with the cross-domain algorithms DTL [Long *et al.*, 2012] and CD-PLSA [Zhuang *et al.*, 2010a], both of which can handle multiple source domains and multiple target domains directly.

The experimental results are shown in Table 2. For each task we record the accuracy values of the two target domains respectively. Again, HIDC exhibits the best performance.

5 Related Work

Transfer Learning arouses vast amount of studies in recently years [Pan and Yang, 2010], and we briefly introduce the most recent and related works to this paper.

The work of transfer learning can be approximately grouped into three categories, namely instance-based approaches [Dai *et al.*, 2007b; Jiang and Zhai, 2007], weighting-based methods [Gao *et al.*, 2008; Dredze *et al.*, 2010], and feature-based algorithms [Dai *et al.*, 2007a; Long *et al.*, 2012; Li *et al.*, 2009; Wang *et al.*, 2011; Zhuang *et al.*, 2010a]. Our model HIDC belongs to the feature-based algorithms, thus here we focus on reviewing works of this category. Dai et al. [Dai *et al.*, 2007a] proposed a Co-Clustering based Classification (CoCC) algorithm, and used the common words for knowledge transfer. Essentially, they only considered the identical concepts. Zhuang et al. [Zhuang *et al.*, 2010a] and Wang et al. [Wang *et al.*, 2011] actually exploited the homogeneous concepts for knowledge transfer. The most related work is Long et al. [Long *et al.*, 2012], where they developed a Dual-Transfer Learning (DTL) framework, and took into account both the identical and homogeneous concepts. However, they did not explicitly consider the distinct concepts. Our model HIDC fully exploits the three kinds of concepts, and the experimental results show the superiority over all compared baselines. It is worth mentioning that considering the distinct concepts is very important and not a trivial work, since HIDC significantly outperforms DTL.

6 Conclusions

In this paper, we deeply analyze the characteristics between the source and target domains, and define three kinds of concepts, i.e., *homogeneous concepts*, *identical concepts* and *distinct concepts*. Along this line, we propose a general cross-domain learning framework based on generative models, which takes into account the three kinds of concepts simultaneously. Then an EM algorithm is developed to derive the solution of our model. Finally, we conduct extensive experiments, no matter the distinct concept is intentionally introduced or not, to demonstrate the superiority of our model.

Acknowledgements

The work is supported by the National Natural Science Foundation of China (No.61175052, 61203297, 60933004, 61035003), National High-tech R&D Program of China (863 Program) (No.2013AA01A606, 2012AA011003), National Program on Key Basic Research Project (973 Program) (No.2013CB329502).

References

[Blitzer *et al.*, 2007] J. Blitzer, M. Dredze, and F. Pereira. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proc. of ACL*, pages 440–447, 2007.

[Boser *et al.*, 1992] B. E. Boser, I. Guyou, and V. Vapnik. A training algorithm for optimal margin classifiers. In *Proc. of the 5th AWCLT*, 1992.

[Dai *et al.*, 2007a] W. Y. Dai, G. R. Xue, Q. Yang, and Y. Yu. Co-clustering based classification for out-of-domain documents. In *Proc. of the 13th ACM SIGKDD*, pages 210–219, 2007.

[Dai *et al.*, 2007b] W. Y. Dai, Q. Yang, G. R. Xue, and Y. Yu. Boosting for transfer learning. In *Proc. of the 24th ICML*, pages 193–200, 2007.

[Dai *et al.*, 2008] W. Y. Dai, Y. Q. Chen, G. R. Xue, Q. Yang, and Y. Yu. Translated learning: Transfer learning across different feature spaces. In *Proc. of the 22nd NIPS*, 2008.

[Dai *et al.*, 2009] 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 ICML*, pages 193–200, 2009.

[Dredze *et al.*, 2010] M. Dredze, A. Kulesza, and K. Crammer. Multi-domain learning by confidence-weighted parameter combination. *Machine Learning*, 79(1):123–149, 2010.

[Gao *et al.*, 2008] J. Gao, W. Fan, J. Jiang, and J. W. Han. Knowledge transfer via multiple model local structure mapping. In *Proc. of the 14th ACM SIGKDD*, pages 283–291, 2008.

[Hofmann, 1999] T. Hofmann. Probabilistic latent semantic analysis. In *Proc.of 15th UAI*, pages 289–296, 1999.

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

[Jiang and Zhai, 2007] J. Jiang and C. X. Zhai. Instance weighting for domain adaptation in nlp. In *Proc. of the 45th ACL*, pages 264–271, 2007.

[Jiho and Choi, 2009] Y. Jiho and S. J. Choi. Probabilistic matrix tri-factorization. In *Proc. of the 2009 IEEE ICASSP*, pages 1553–1556, 2009.

[Joachims, 1999] T. Joachims. Transductive inference for text classification using support vector machines. In *Proc. of the 16th ICML*, 1999.

[Li *et al.*, 2009] T. Li, V. Sindhwani, C. Ding, and Y. Zhang. Knowledge transformation from for cross-domain sentiment classification. In *Proc. of the 32st SIGIR*, pages 716–717, 2009.

[Long *et al.*, 2012] M. Long, J. Wang, G. Ding, W. Cheng, X. Zhang, and W. Wang. Dual transfer learning. In *Proc. of the 12th SIAM SDM*, 2012.

[Luo *et al.*, 2008] P. Luo, F. Z. Zhuang, H. Xiong, Y. H. Xiong, and Q. He. Transfer learning from multiple source domains via consensus regularization. In *Proc. of the 17th ACM CIKM*, pages 103–112, 2008.

[Pan and Yang, 2010] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE TKDE*, 22(10):1345–1359, 2010.

[Wang *et al.*, 2011] H. Wang, H. Huang, F. Nie, and C. Ding. Cross-language web page classification via dual knowledge transfer using nonnegative matrix tri-factorization. In *Proc. of the 34th ACM SIGIR*, pages 933–942, 2011.

[Xue *et al.*, 2008] 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 SIGIR*, pages 627–634, 2008.

[Zhuang *et al.*, 2010a] F. Zhuang, P. Luo, Z. Shen, Q. He, Y. Xiong, Z. Shi, and H. Xiong. Collaborative dual-plsa: mining distinction and commonality across multiple domains for text classification. In *Proc. of the 19th ACM CIKM*, pages 359–368, 2010.

[Zhuang *et al.*, 2010b] 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 10th SIAM SDM*, pages 13–24, 2010.