Instance-Based Domain Adaptation in NLP via In-Target-Domain Logistic Approximation

Rui Xia[1], Jianfei Yu[1], Feng Xu[2], and Shumei Wang[1]
[1]School of Computer Science and Engineering, Nanjing University of Science and Technology, China
[2]School of Economics and Management, Nanjing University of Science and Technology, China
{rxia.cn, yujianfei1990, xufeng.breeze}@gmail.com, hwasm@njust.edu.cn

Abstract

In the field of NLP, most of the existing domain adaptation studies belong to the feature-based adaptation, while the research of instance-based adaptation is very scarce. In this work, we propose a new instance-based adaptation model, called in-target-domain logistic approximation (ILA). In ILA, we adapt the source-domain data to the target domain by a logistic approximation. The normalized in-target-domain probability is assigned as an instance weight to each of the source-domain training data. An instance-weighted classification model is trained finally for the cross-domain classification problem. Compared to the previous techniques, ILA conducts instance adaptation in a dimensionality-reduced linear feature space to ensure efficiency in high-dimensional NLP tasks. The instance weights in ILA are learnt by leveraging the criteria of both maximum likelihood and minimum statistical distance. The empirical results on two NLP tasks including text categorization and sentiment classification show that our ILA model has advantages over the state-of-the-art instance adaptation methods, in cross-domain classification accuracy, parameter stability and computational efficiency.

Introduction

For many NLP tasks, e.g., text categorization, sentiment classification, etc., it is nowadays very easy to obtain a large collection of labeled data from different domains in the vast amount of Internet texts. But not all of them are useful for training a desired target-domain classifier. Thus, it is necessary for us to employ an instance adaptation technique to identify the most important training instances, and increase their weights in the training process. However, to the best of our knowledge, most existing work for domain adaptation in NLP employs feature-based adaptation, while the research of instance-based adaptation is very scarce (Jiang and Zhai, 2007; Pan and Yang, 2010; Xia et al., 2013a)·

The instance adaptation methods were mainly proposed by the machine learning community in the past. In machine learning, "instance adaptation" is also termed "covariate shift" or "instance selection bias", where the key problem is density ratio estimation (DRE). Series of kernel-based methods were proposed to solve the DRE problem (Shimodaira, 2000; Huang et al., 2007; Sugiyama et al., 2007; Tsuboi et al., 2008; Kanamori et al., 2009). Among them, the KLIEP algorithm (Sugiyama et al., 2007) is the representative one. It estimates the density ratio based on a linear model in a Gaussian kernel space.

However, the kernel-based methods are mostly designed under tasks of low-dimensional continuous distributions. It is hard to apply them directly to tasks of high-dimensional discrete distributions. E.g., if KLIEP is applied to such tasks, it is difficult to choose a suitable kernel function. The kernel function mapping in high-dimensional feature space is also computationally impractical.

In this work, we propose a new instance adaptation model, called in-target-domain logistic approximation (ILA), to adapt the source-domain training data to the target domain by a logistic approximation. In ILA, instance adaptation is conducted in a linear feature space, rather than a complex kernel space. A domain-sensitive feature selection method is proposed furthermore to reduce the dimensionality of the linear feature space. Both make ILA efficient for high-dimensional NLP tasks.

More recently, Xia et al. (2013b) proposed an instance weighting approach via PU learning (PUIW) for domain adaptation in sentiment classification. Although PUIW is applicable to high-dimensional NLP tasks, the instance weights are learnt by two separated steps in PUIW. The instance weight learning is not efficient, and the adaptation performance depends heavily on the preset value of the calibration parameter. In ILA, the instance weights are

estimated by leveraging the criteria of both maximum likelihood and minimum statistical distance in one single model. It makes ILA more stable in parameter sensitivity.

We evaluate our ILA algorithm on ten datasets of two NLP tasks including cross-domain text categorization and sentiment classification. The empirical results show that ILA is superior to the state-of-the-art instance adaptation methods, in classification accuracy, parameter stability and computational efficiency.

Related Work

While the feature-based adaptation has been sufficiently studied in the field of NLP (Daume III, 2007; Blitzer et al., 2007; Pan et al., 2008; Pan et al., 2010; Glorot et al., 2011; Duan et al., 2012), the work of instance-based adaptation is relatively scarce. In this work, we focus on instance-based adaptation.

In the machine learning community, instance adaptation is also known as the "covariate shift" or "instance selection bias" (Zadrozny, 2004). There the key issue is the density ratio estimation (DRE). The estimated density ratio could then be used to generate weighted training samples for statistical machine learning. There were series of kernel-based methods to solve the DER problem. For example, Shimodaira (2000), Dudik et al. (2005) and Huang et al. (2007) utilized kernel density estimation, maximum entropy density estimation, and kernel mean matching respectively. Sugiyama et al., (2007) proposed a KLIEP algorithm to directly estimate the density ratio by using a linear model in a Gaussian kernel space. Parameters were learnt by minimizing the K-L divergence between the true and approximated distributions. The least square criterion was also studied in (Kanamori et al., 2009). Tsuboi et al. (2008) extended KLIEP by employing a log-linear model instead of the linear model. It made KLIEP feasible in the setting of large-scale test dataset, yet with low-dimensional feature space.

However, it is hard to apply these kernel-based methods to the NLP tasks of high-dimensional discrete distributions directly. The ILA model proposed here uses a logistic approximation for instance adaptation in a dimensionality-reduced linear space, without kernel function mapping. It makes instance adaptation applicable to high-dimensional NLP tasks.

Bickel et al. (2007) utilized a logistic regression model to learn the density ratio together with the classification parameters, under the multi-task learning framework. Its aim is to maximize the likelihood of data in both domains. By contrast, the goal in ILA is to use the instance-weighted source-domain labeled data to maximize the likelihood of the data in the target domain.

Recently, Xia et al. (2013b) proposed instance selection and instance weighting algorithm via PU learning for domain adaptation in sentiment classification. Instance weights were learnt by two separate steps. Each training sample is assigned with an in-target-domain probability at first; the calibrated probabilities were then used as weights to train an instance-weighted sentiment classifier. In comparison, the instance weights in ILA are estimated in one single model.

Problem Formalization

To facilitate the following discussion, we introduce some notations at first. Let $p(\mathbf{x})$, $p(y)$ and $p(y|\mathbf{x})$ respectively denote the instance, class and posterior probability, where $\mathbf{x} \in \mathcal{X}$ is the feature vector, and $y \in \mathcal{Y}$ is the class label. The subscript s and t denote the source and target domain.

Let θ be the parameter of a classification model. Since labeled data are not available in the target domain, the goal of instance adaptation is to use the source-domain labeled data as an approximate, to maximize the likelihood of the data in the target domain:

$$\begin{aligned}\theta^* &= \arg\max_\theta \int_{\mathcal{X}} \tilde{p}_t(\mathbf{x}) \sum_{y\in\mathcal{Y}} \tilde{p}_t(y|\mathbf{x}) \log p(\mathbf{x}, y|\theta) d\mathbf{x} \\ &= \arg\max_\theta \int_{\mathcal{X}} \frac{\tilde{p}_t(\mathbf{x})}{p_s(\mathbf{x})} p_s(\mathbf{x}) \sum_{y\in\mathcal{Y}} p_s(y|\mathbf{x}) \log p(\mathbf{x}, y|\theta) d\mathbf{x}\end{aligned}$$

where $\tilde{p}_t(\mathbf{x})$ and $\tilde{p}_t(y|\mathbf{x})$ denote the approximated target-domain distributions, which are adapted from the source domain. [1]

We use $w(\mathbf{x}) = \frac{\tilde{p}_t(\mathbf{x})}{p_s(\mathbf{x})}$ to denote the instance weight. The empirical form of the above problem is:

$$\theta^* = \arg\max_\theta \frac{1}{N_s} \sum_{n=1}^{N_s} w(\mathbf{x}_n) \log p(\mathbf{x}_n, y_n|\theta) \qquad (1)$$

where N_s is the size of source-domain training set.

Therefore, the key problem in instance adaptation is the estimation of the instance weight $w(\mathbf{x}) = \frac{\tilde{p}_t(\mathbf{x})}{p_s(\mathbf{x})}$.

The Instance Adaptation Model

In this work, we propose an instance adaptation approach, called in-target-domain logistic approximation (ILA).

In-target-domain Logistic Approximation

In ILA, it is assumed that a target-domain instance $\mathbf{x}$ is generated by the following instance adaptation process:

1) An instance $\mathbf{x}$ is drawn by first sampling $\mathbf{x}$ from the source-domain distribution $p_s(\mathbf{x})$;
2) An in-target-domain selector then adapts $\mathbf{x}$ to the target domain, based on a logistic approximation.

[1] The second equation holds because it is assumed in instance-based adaptation that $\tilde{p}_t(y|\mathbf{x}) \approx p_s(y|\mathbf{x})$ (Pan and Yang, 2010).

Under this assumption, the approximated target-domain instance distribution can be formulized as:

$$\tilde{p}_t(\mathbf{x}) = \alpha \cdot \frac{1}{1+e^{-\beta^{\mathrm{T}}\mathbf{x}}} p_s(\mathbf{x}) \quad (2)$$

where α is a normalization factor making $\tilde{p}_t(x)$ a valid probability; β is the feature weight. Note that α and β are parameters of the instance adaptation model. They should be distinguished from the parameter θ of the classification model in Equation (1).

The normalized in-target-domain probability

$$w(\mathbf{x}) = \frac{\alpha}{1+e^{-\beta^{\mathrm{T}}\mathbf{x}}} \quad (3)$$

will be used as instance weights for training an instance-weighted classification model after instance adaptation.

Instance Adaptation Parameter Learning

There are two different types of criteria that can be used to learn the instance adaptation parameters.

Maximum Likelihood (ML): On one hand, we can view the in-target-domain selector as a binary classification problem (i.e., a logistic regression model), where the class labels are the "target domain" and "source domain". Parameters are learnt to best distinguish data of two different domains. For this purpose, we define the negative log-likelihood function as:

$$J_{ml} = -\frac{1}{N}\left(\sum_{i=1}^{N_t} \log \frac{1}{1+e^{-\beta^{\mathrm{T}}\mathbf{x}_i}} + \sum_{j=1}^{N_s} \log \frac{e^{-\beta^{\mathrm{T}}\mathbf{x}'_j}}{1+e^{-\beta^{\mathrm{T}}\mathbf{x}'_j}}\right) (4)$$

where $\mathbf{x}'$ denotes the source-domain training sample, N_s and N_t respectively denote the size of the source and target domain training set, and $N = N_s + N_t$.

According to Equation (3), the posterior probability of a sample belonging to the target domain is proportional to the instance weight in instance adaptation. Therefore, by maximizing Equation (4), the samples with higher target-domain probability will receive relatively larger weights in instance adaptation.

In fact, the ML criterion was originally used in PUIW (Xia et al., 2013b), where a semi-supervised target/source domain classifier was learnt based on EM algorithm. But in PUIW the in-target-domain probability should be calibrated before serving as the instance weight. By contrast, the instance weights in ILA are estimated in one single model, based on a combined cost function.

Minimum Statistical Distance (MSD): On the other hand, we can learn the parameters by minimizing the statistical distance between the true target-domain distribution $p_t(\mathbf{x})$ and the approximated one $\tilde{p}_t(\mathbf{x})$. Sugiyama et al. (2007) proposed a Kullback-Leibler (K-L) importance estimation procedure (i.e., KLIEP) under a linear instance adaptation model.

Here, we will derive the minimum K-L divergence criterion under ILA:

$$\begin{aligned} KL(p_t||\tilde{p}_t) &= \int_{\mathcal{X}} p_t(\mathbf{x}) \log \frac{p_t(\mathbf{x})}{\tilde{p}_t(\mathbf{x})} d\mathbf{x} \\ &= KL(p_t||p_s) - \int_{\mathcal{X}} p_t(\mathbf{x}) \log \frac{\alpha}{1+e^{-\beta^{\mathrm{T}}\mathbf{x}}} d\mathbf{x}. \end{aligned}$$

Note that the first term is the K-L divergence of $p_t(\mathbf{x})$ and $p_s(\mathbf{x})$. It is independent of α and β, and can be ignored in optimization:

$$\arg\min_{\alpha,\beta} KL(p_t||\tilde{p}_t) = \arg\min_{\alpha,\beta} -\int_{\mathcal{X}} p_t(\mathbf{x}) \log \frac{\alpha}{1+e^{-\beta^{\mathrm{T}}\mathbf{x}}} d\mathbf{x}.$$

We add the constraint that $\tilde{p}_t(\mathbf{x})$ is a valid probability

$$\int_{\mathcal{X}} \tilde{p}_t(x) dx = 1,$$

and take the empirical form of the optimization problem:

$$\begin{aligned} \min_{\alpha,\beta} \quad & -\frac{1}{N_t}\sum_{i=1}^{N_t} \log \frac{\alpha}{1+e^{-\beta^{\mathrm{T}}\mathbf{x}_i}} \\ s.t. \quad & \frac{1}{N_s}\sum_{j=1}^{N_s} \frac{\alpha}{1+e^{-\beta^{\mathrm{T}}\mathbf{x}'_j}} = 1. \end{aligned}$$

Such an equality constrained problem was optimized by gradient descent with feasibility satisfaction in KLIEP.

In ILA, the problem can become unconstrained by solving α from the equality constraint and plugging that back into the cost function. This leads to optimization efficiency in comparison with KLIEP.

After removing the constant term, we get the final minimum statistical distance cost function:

$$J_{msd} = \frac{1}{N_t}\sum_{i=1}^{N_t} \log(1+e^{-\beta^{\mathrm{T}}\mathbf{x}_i}) + \log \sum_{j=1}^{N_s} \frac{1}{1+e^{-\beta^{\mathrm{T}}\mathbf{x}'_j}}. \quad (5)$$

The Combined Cost Function: In density ratio estimation, it is reasonable to learn the instance weight by minimizing the statistical distance such as K-L divergence. However, in instance adaptation for cross-domain classification, the criterion sometimes tends to be arbitrary due to over-fitting. Blindly minimizing the statistical distance may encourage the system to assign particularly large weights to the most target-domain-relevant instances. If the instance adaptation assumption (e.g., $\tilde{p}_t(y|\mathbf{x}) \approx p_s(y|\mathbf{x})$) does not hold in these samples, the cross-domain classification performance will be severely hurt. By contrast, the ML criterion seems to be more moderate.

Therefore, we propose a combined cost function to leverage two different types of criteria for learning the parameters in ILA:

$$J = \lambda J_{msd} + (1-\lambda) J_{ml} \quad (6)$$

where $\lambda \in [0,1]$ is a tradeoff parameter. When $\lambda = 0$, it becomes the ML criterion; when $\lambda = 1$, it becomes the MSD criterion.

Gradient Descent Optimization: Since both J_{ml} and J_{msd} are unconstrained, we can easily use the gradient descent method to minimize J. The gradients of J_{ml} and J_{msd} are as follows:

$$\frac{\partial J_{msd}}{\partial \beta} = \frac{1}{\sum_{j=1}^{N_s} \delta(\beta^{\mathrm{T}} \mathbf{x}'_j)} \sum_{j=1}^{N_s} \delta(\beta^{\mathrm{T}} \mathbf{x}'_j) \left(1 - \delta(\beta^{\mathrm{T}} \mathbf{x}'_j)\right) \mathbf{x}'_j - \frac{1}{N_t} \sum_{i=1}^{N_t} \left(1 - \delta(\beta^{\mathrm{T}} \mathbf{x}_i)\right) \mathbf{x}_i$$

$$\frac{\partial J_{ml}}{\partial \beta} = \frac{1}{N_s + N_t} \left(\sum_{i=1}^{N_t} 1 - \delta(\beta^{\mathrm{T}} \mathbf{x}_i) \mathbf{x}_i - \sum_{j=1}^{N_s} \delta(\beta^{\mathrm{T}} \mathbf{x}'_j) \mathbf{x}'_j \right)$$

where $\delta(\cdot)$ denotes the sigmoid function.

Instance Adaptation Feature Selection

Furthermore, we propose a domain-sensitive feature selection technique, to reduce the dimension of the linear feature space in ILA.

Information gain (IG) has been identified as one of the best feature selection methods (Yang and Pedersen, 1997) in document categorization. Motivated by that, we use IG to calculate the dependence of features and domains. Note that here we aim to select domain-sensitive features. Thus, we modify the standard IG to calculate the relevance between a term x_k and the domain indicator variable d (rather than the class label y):

$$\begin{aligned} IG(x_k) = & - \sum_{l \in \{0,1\}} p(d=l) \log p(d=l) \\ & + p(x_k) \sum_{l \in \{0,1\}} p(d=l|x_k) \log p(d=l|x_k) \\ & + p(\bar{x}_k) \sum_{l \in \{0,1\}} p(d=l|\bar{x}_k) \log p(d=l|\bar{x}_k). \end{aligned}$$

where $d=1$ denotes the target domain, $d=0$ denotes the source domain. The top-ranked terms will be selected as features in our ILA algorithm. In the experimental study, we will discuss the effects of feature selection in ILA.

Instance-weighted Classification Model

So far we have introduced the instance adaptation model. Once the parameter α and β have been learnt, we first use Equation (3) to calculate the instance weight for each source-domain training sample. Then, we train an instance-weight classification model based on Equation (1) for the cross-domain classification problem.

Standard classification models, such as Naïve Bayes, MaxEnt and SVMs, can all be extended to an instance-weighted version, by incorporating the instance weights into the training process. In this work, we only adopt the instance-weighted naïve Bayes (IWNB) model. Details of IWNB can be found in (Xia et al., 2013b).

Experimental Study

Experimental Settings and Datasets

To fully evaluate the performance of ILA, we conduct the domain adaptation experiments on two different NLP tasks: 1) text categorization; 2) sentiment classification.

For text categorization, we employ the 20 Newsgroups dataset[2] for experiments. It contains seven top categories, and there are 20 subcategories under the top categories. We follow the experimental settings in (Dai et al., 2007) for domain adaptation. That is, we select four top categories ("com", "rec", "sci" and "talk") as the class labels, and generate source and target domains based on subcategories. For instance, "med" and "space" are two subcategories under "sci", and "guns" and "misc" are two subcategories under "talk". The datasets are split in such a way that "med" and "guns" are used as the source domain data, and "space" and "misc" are used as the target domain data.

For sentiment classification, we follow the datasets and experimental settings used by (Xia et al., 2013b). That is, the Movie Review dataset[3] is used as the source domain, and each of the Multi-domain sentiment datasets[4] (Book, DVD, Electronics, and Kitchen) serves as the target domain. We randomly choose 200 labeled data from the target domain, and mix them with 2000 source-domain labeled data[5] to construct a domain-mixed training dataset. The remaining data in the target domain is used as the test set.

In both of the two tasks, unigrams and bigrams with term frequency no less than 4 are used as features for classification. We randomly repeat the experiments for 10 times, and report the average results in Table 1. The tradeoff parameter λ is set to be 0.7 in text categorization and 0.6 in sentiment classification. The percentage in instance adaptation feature selection is set to be 30% and 50% in text categorization and sentiment classification, respectively. To avoid the over-fitting problem mentioned in the MSD criterion, we set the maximum iteration steps in gradient descent optimization as 30. The paired *t*-test (Yang and Liu, 1999) is employed for significance testing.

[2] http://qwone.com/~jason/20Newsgroups/
[3] http://www.cs.cornell.edu/people/pabo/movie-review-data/
[4] http://www.cs.jhu.edu/~mdredze/datasets/sentiment/
[5] It is designed to test if the instance adaptation approach could identify the hidden target-domain-relevant samples and make full use of them.

Task	Dataset	K-L Divergence	No Adaptation	KLIEP		PUIS	PUIW	ILA
				Linear	Gaussian			
Text categorization	sci vs com	28.3	0.602	0.504	0.624	0.602	0.619	**0.630**
	talk vs com	18.5	0.908	0.910	0.922	0.909	0.907	**0.959**
	sci vs talk	29.3	0.852	0.851	0.855	0.880	0.863	**0.921**
	rec vs sci	28.3	0.651	0.593	0.652	0.689	**0.742**	**0.742**
	rec vs com	20.6	0.900	0.693	0.910	0.901	0.911	**0.922**
	talk vs rec	35.3	0.820	0.821	0.821	0.821	0.834	**0.837**
	Avg.	26.7	0.788	0.729	0.797	0.800	0.813	**0.835**
Sentiment classification	movie → book	4.06	0.756	0.737	0.768	0.757	0.774	**0.780**
	movie → dvd	2.12	0.762	0.738	0.783	0.762	0.782	**0.796**
	movie → elec	13.4	0.697	0.673	0.741	0.726	0.750	**0.768**
	movie → kitchen	13.4	0.709	0.626	0.759	0.743	0.777	**0.785**
	Avg.	8.25	0.731	0.694	0.763	0.747	0.771	**0.783**

Table 1: Domain adaptation performance of different systems on two NLP tasks. In text categorization, "A vs B" means that the top category A and B are used as class labels, and subcategories under the top categories are used to generate the source and target domain datasets. In sentiment classification, "A → B" denote that we use dataset A as the source domain, and B as the target domain.

Compared Systems

The following systems are implemented for comparison with our ILA model:

1) **No-Adaptation**: the standard machine learning method using all training samples;
2) **KLIEP-Linear**: the KLIEP model (Sugiyama et al., 2007) using a linear kernel;
3) **KLIEP-Gaussian**: the KLIEP model using a Gaussian kernel;
4) **PUIS**: the instance selection model proposed by (Xia et al., 2013b) via PU learning;
5) **PUIW**: the instance weighting model proposed by (Xia et al., 2013b) via PU learning.

Domain Adaptation Performance

In Table 1, we report the domain adaptation performance of the six evaluated systems on text categorization and sentiment classification.

Text categorization: It can be seen that KLIEP-Linear fails in instance adaptation. It is even 5.9% lower than the No-Adaptation baseline. KLIEP-Gaussian is much better, but the improvement is very limited (0.9%). PUIS shows comparable performance to KLIEP-Gaussian (0.800 vs. 0.797), which is also less efficient. PUIW is shown to be quite efficient in instance adaptation. It gains a 2.5% increase against the No-Adaptation baseline.

By contrast, our ILA model yields the best performance (0.835). It outperforms the No-Adaptation, KLIEP-Linear, KLIEP-Gaussian, PUIS and PUIW systems 4.7, 10.6, 3.8, 3.5 and 2.2 percentages, respectively. All improvements are significant according to the paired *t*-test (*p*-value<0.05), except for the "rec vs sci" dataset compared to PUIW.

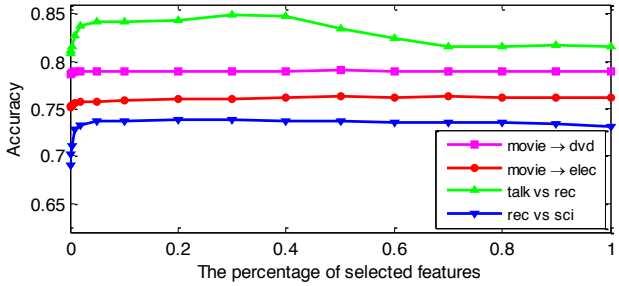

Figure 1: The effect of domain-sensitive feature selection in ILA. The x-axis denotes the percentage of selected features; the y-axis denotes the domain adaptation accuracy.

Sentiment classification: The observation is similar to that in text categorization. KLIEP-Linear still fails in instance adaptation; while this time KLIEP-Gaussian behaves more efficient. It improves the No-Adaptation baseline 3.2%, and beats PUIS (0.763 vs. 0.747), but is still slightly lower than PUIW (0.763 vs. 0.771). The performance of our ILA model is still sound (0.783). It outperforms No-Adaptation, KLIEP-Linear, KLIEP-Gaussian, PUIS and PUIW 5.2, 8.9, 2.0, 3.6 and 1.2 percentages, respectively. All of the improvements are significant (*p*-value<0.05).

Effect of Instance Adaptation Feature Selection

In this subsection, we discuss the effect of feature selection in instance adaptation. Due to space limitation, we only present the results on four datasets in Figure 1. The same conclusions can be drawn from the other datasets.

It can be observed across four datasets that, using only 1-10% (500-5,000) features can obtain a comparable (or even better) performance than all features (around 50,000) in ILA. It suggests that our ILA model can work efficiently in a dimensionality-reduced linear feature space.

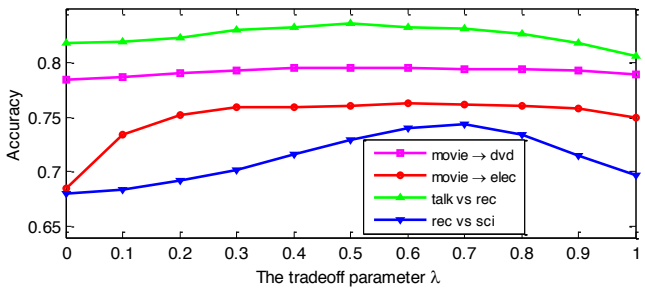

Figure 2: Sensitivity of the tradeoff parameter in ILA. The x-axis denotes the value of the tradeoff parameter λ in Equation (6).

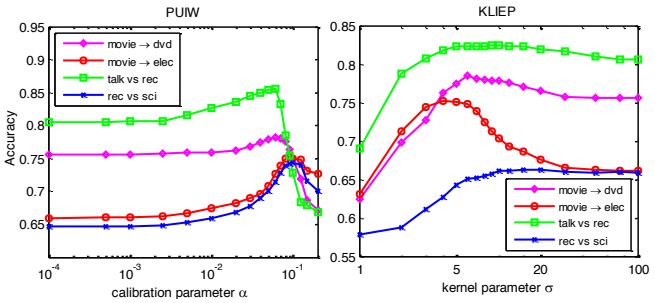

Figure 3: Parameter stability of PUIW and KLIEP-Gaussian. The x-axis denotes the value of the calibration parameter α in PUIW (left), and the kernel parameter δ in KLIEP-Gaussian (right).

Sensitivity to the Tradeoff Parameter λ

We further discuss the sensitivity of the tradeoff parameter λ in ILA. In Figure 2, we still only present the results on four datasets. Similar conclusions can be drawn from the other datasets.

As shown in Figure 2, neither the ML criterion ($\lambda = 0$) nor the MSD criterion ($\lambda = 1$) yields the best instance adaptation performance. In all of the four datasets, the tradeoff parameter λ is quite stable. The best accuracy can be obtained when λ is located at 0.5 to 0.7. It suggests that the ML and MSD criteria have distinct strength, and a combination of them is reasonable for instance adaptation.

Comparison of Parameter Stability

In this subsection, we compare ILA, KLIEP-Gaussian and PUIW in parameter stability. In Figure 3, we report the sensitivity of the calibration parameter α in PUIW, and the kernel function parameter δ in KLIEP-Gaussian.

It can be seen that the performances of both PUIW and KLIEP-Gaussian change dramatically as their parameters change. In PUIW, the domain adaptation accuracy drops 8% when α changes from 0.1 to 0.01. The change of δ from 5 to 1 in KLIEP-Gaussian may cause a decline of more than 10%. Moreover, the best parameters are not consistent across different datasets. For example, KLIEP-Gaussian obtains the best accuracy on four datasets when σ is 5, 8, 15 and 20, respectively.

By contrast, our ILA model tends to be more moderate and stable in parameter tuning.

Comparison of Computational Efficiency

Finally, we compare the computational efficiency of three models. We implement all three algorithms with Python, and run the experiments on a server with a 2.2GHz Intel Xeon Processor and 4GB RAM.

In Table 2, we report the average computational time for running KLIEP-Gaussian, PUIW and ILA on two tasks, respectively. Observed from the results, the computational cost of KLIEP-Gaussian is much higher than that of PUIW and ILA. Note that in KLIEP-Gaussian, we have already applied domain-sensitive feature selection before kernel function mapping, otherwise the computational times will be much longer. The computational cost of PUIW is about 1.5 to 2 times that of ILA.

Task	KLIEP-Gaussian #1000	KLIEP-Gaussian #100	PUIW	ILA
Text categorization	19346s	2121s	241s	161s
Sentiment classification	5219s	482s	184s	97s

Table 2: Comparisons of computational efficiency. "#1000" and "#100" mean that the number of centers used in the Gaussian kernel function is 1000 and 100 respectively.

It indicates that our ILA approach outperforms the state-of-the-art approaches in computational efficiency.

Conclusions and Future Work

We propose an instance adaptation model called in-target-domain logistic approximation (ILA) for high-dimensional domain adaptation tasks. ILA works in a dimensionality-reduced linear feature space, and learns the instance weights by leveraging the criteria of both maximum likelihood and minimum statistical distance. In comparison with the existing instance domain adaptation approaches, ILA has certain advantages in cross-domain classification accuracy, parameter stability and computational efficiency.

One of the shortcomings in ILA is that it only models the adaptation of marginal distribution ($p_s(\mathbf{x}) \rightarrow \tilde{p}_t(\mathbf{x})$), but ignore the adaptation of $p_s(y|\mathbf{x}) \rightarrow p_t(y|\mathbf{x})$. This may not hold for real-world applications. We plan to conduct an in-depth study regarding to this point in our future work.

Acknowledgments

The research work is supported by the Natural Science Foundation of China (61305090), the Jiangsu Provincial Natural Science Foundation of China (BK2012396), and the Research Fund for the Doctoral Program of Higher Education of China (20123219120025).

References

Bickel, S.; Brückner, M.; and Scheffer, T. 2007. Discriminative learning for differing training and test distributions. In *Proc*. *of ICML*.

Blitzer, J.; Dredze, M.; and Pereira, F. 2007. Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification. In *Proc*. *of ACL*.

Dai, W.; Xue, G.; Yang, Q.; and Yu, Y. 2007. Transferring Naive Bayes Classifiers for Text Classification. In *Proc*. *of AAAI*.

Daume III H. 2007. Frustratingly Easy Domain Adaptation. In *Proc*. *of ACL*.

Duan, L. ; Xu, D. ; and Tsang, I. W. 2012. Learning with Augmented Features for Heterogeneous Domain Adaptation. In *Proc*. *of ICML*.

Dudik, M.; Schapire, R.; and Yu, P. S. 2005. Correcting sample selection bias in maximum entropy density estimation. In *NIPS*.

Glorot, X.; Bordes, A.; and Bengio, Y. 2011. Domain Adaptation for Large-Scale Sentiment Classification: A Deep Learning Approach. In *Proc*. *of ICML*.

Huang, J.; Smola, A.; Gretton, A.; Borgwardt, K.; and Schölkopf, B. 2007. Correcting sample selection bias by unlabeled data. In *NIPS*.

Jiang, J., and Zhai, C. 2007. Instance weighting for domain adaptation in NLP. In *Proc*. *of ACL*.

Kanamori, T.; Hido, S.; and Sugiyama, M. 2009. A least-squares approach to direct importance estimation. *Journal of Machine Learning Research*, 10:1391–1445.

Pan, S. J.; Tsang, I. W.; Kwok, J. T.; and Yang, Q. 2008. Domain Adaptation via Transfer Component Analysis. In *Proc of IJCAI*.

Pan, S. J., and Yang, Q. 2010. A survey on transfer learning. *IEEE Trans*. *Knowledge and Data Engineering*, 22(10): 1345–1359.

Pan, S. J.; Ni, X.; Sun, J.; Yang, Q.; and Chen, Z. 2010. Cross-domain sentiment classification via spectral feature alignment. In *Proc*. *of WWW*.

Shimodaira, H. 2000. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90:227–244, 2000.

Sugiyama, M.; Nakajima, S.; Kashima, H.; Bunau, P. V.; and Kawanabe, M. 2007. Direct importance estimation with model selection and its application to covariate shift adaptation. In *NIPS*.

Tsuboi, Y.; Kashima, H.; Hido, S.; Bickel, S.; and Sugiyama, M. 2008. In *Prof*. *of SDM*.

Xia, R; Zong, C; Hu, X.; and Cambria, E. 2013a. Feature Ensemble plus Sample Selection: Domain Adaptation for Sentiment Classification. *IEEE Intelligent Systems*, 28(3): 10–18.

Xia, R.; Hu, X.; Lu, J.; Yang, J.; and Zong, C. 2013b. Instance Selection and Instance Weighting for Cross-domain Sentiment Classification via PU Learning. In *Proc of IJCAI*.

Yang, Y., and Liu, X. 1999. A re-examination of text categorization methods. In *Proc*. *of SIGIR*.

Yang, Y., and Pedersen, J. O. 1997. A Comparative Study on Feature Selection in Text Categorization. In *Proc*. *of ICML*.

Zadrozny, B. 2004. Learning and evaluating classifiers under sample selection bias. In *Proc*. *of ICML*.