Predicting Sentiment using Tranfer Learning

Anastasia Krithara, George Giannakopoulos, George Paliouras, George Petasis and Vangelis Karkaletsis
Institute of Informatics and Telecommunications,
National Center for Scientific Research (NCSR) "Demokritos",
Athens, Greece
{akrithara, ggianna, paliourg, petasis, vangelis}@iit.demokritos.gr

Abstract

A new transfer learning method is presented in this paper, addressing the task of sentiment analysis across domains.The proposed approach is a transfer learning variant of the Probabilistic Latent Semantic Analysis (PLSA) model that we name KLIEP-PLSA. The approach captures the difference of the word distributions between the different domains. We perform experiments over well known datasets and show the promising results that we obtained with the new method.

1 Introduction

Machine learning technologies have already achieved significant success in many knowledge engineering areas including classification, regression and clustering. However, the vast majority of machine learning algorithms operate under a basic assumption: both the training and test data should use the same feature space, and follow the same distribution, suggesting that both should originate from the same thematic domain. When the distribution changes, the models must be re-generated from newly collected data.

In many real world applications, it is expensive or impossible to collect the needed training data and rebuild the models. Knowledge transfer would greatly improve the performance of learning by avoiding expensive data-labeling efforts. In recent years, *transfer learning* has emerged as a new learning framework to address this problem. It tries to extract knowledge from previous experience and apply it on new learning domains or tasks.

As an example, we may want to learn a sentiment analysis method for one domain, but we may not have such examples pre-annotated with appropriate sentiment. However, we may be given plenty of annotated examples from a different - but somehow related - domain.

Existing transfer learning approaches can be categorized into three main types [Pan and Yang, 2010], based on the characteristics of the source and target domains and tasks:

1. *Inductive transfer:* The target task is different from the source task and some labeled data in the target domain are required. For document classification, two tasks are considered different if either the label sets are different in the two domains, or the source and target documents are very imbalanced in terms of user-defined classes. Depending on the availability of labeled data in the source domain, we distinguish two subcategories:

   - Labeled data in the source domain are available. This setting is similar to multitask learning.
   - No labeled data in the source domain are available. This setting is similar to self-taught learning.

   Most existing approaches of this type focus on the former subcategory.

2. *Transductive transfer learning setting:* The source and target tasks are the same, while the source and target domains differ. For document classification, two domains are considered different if either the term features are different, or their marginal distributions are different. No labeled data for the target domain are available, while labeled data are available for the source domain.

3. *Unsupervised transfer learning:* Similar to inductive transfer learning, the target task is different from but related to the source task. However, the unsupervised transfer learning focuses on solving unsupervised learning tasks in the target domain, such as clustering. There are no labeled data available in either the source or the target domains.

Transfer learning researches three main central problems [Zhang and Shakya, 2009]: 1) how to extract the prior knowledge that is related, 2) how to represent the knowledge, and 3) how to apply the knowledge in the new learning task. Domain adaptation is a sub-category of transfer learning, where [Pan and Yang, 2010]:

1. The source and target domains are different, but related.
2. The source and target tasks are the same (i.e. classification or regression).
3. Labelled examples are available for the source domain.
4. Only unlabeled examples are available for the target domain.

In this paper, we propose a novel approach for the task of transfer learning. We focus on the task of sentiment analysis, but the approach can be applied to any problem that can be expressed as a classification task. Our method captures the difference of word distributions in the different domains. In particular, an extension of the PLSA method [Hofmann, 2001], which incorporates instance weights, is used. The instance weights are calculated using the KLIEP approach, an algorithm which directly estimate the ratio of two density functions without going through density estimation [Sugiyama *et al.*, 2007].

The rest of the paper is organized as follows: in section 2 related work is presented, where our method is compared to existing approaches. In section 3 presents the extension of the PLSA algorithm using weights. Section 4 presents the evaluation performed. Finally, section 5 concludes this paper and presents some future directions.

2 Related work

Despite its importance, the transfer learning problem only gained sufficient attention in the machine learning community recently. There have been a number of studies on solving specific transfer learning problems or addressing the problem from various perspectives. However, transfer learning is not yet completely understood, and there are no dominating methods that are used widely. The task of transfer learning can be defined as follows: given a source domain D_S, a source task T_S, a target domain $D_T \neq D_S$, and a target task T_T, transfer learning aims to learn a function f_T that accomplishes task T_T, by exploiting knowledge derived from D_S and T_S. A fairly recent overview of the area of transfer learning is given in the survey of [Pan and Yang, 2010], including the definition of transfer learning, its relation to traditional machine learning, a categorisation of transfer learning approaches, and practical applications of transfer learning. More recent approaches that target the task of domain adaptation can be found on the ACL 2010 Workshop on Domain Adaptation for Natural Language Processing (DANLP 2010) [III *et al.*, 2010]. Below, we present some of the approaches in the literature which fall into the two first types of transfer learning, that are most relevant to our work.

Inductive transfer learning TrAdaBoost [Dai *et al.*, 2007b] is an extension of the AdaBoost algorithm. TrAdaBoost assumes that the source and target domain data use exactly the same set of features and labels, but the conditional probability distributions between the domains are different. It also assumes that there are labeled data in both source and target domain data. It attempts to iteratively reweight the source domain data to reduce the effect of the "bad" source data while encouraging the "good" source data to contribute more for the target domain. In the same vein, a heuristic method was proposed in [Jiang and Zhai, 2007], in order to remove "misleading" training examples from the source domain based on the difference between conditional probabilities between domains. Some approaches in inductive transfer learning, try to find new feature representations in order to minimize domain divergence. For example, in [Lee *et al.*, 2007], a convex optimization algorithm for this scope is presented. The idea is to simultaneously learn metapriors and feature weights from an ensemble of related prediction tasks. The metapriors can be transferred among different tasks. Another method [Daume *et al.*, 2010] uses features from source and target domains to construct an augmented feature space. However, despite its simplicity, a formal theoretical analysis is clearly missing. Some other approaches try to take advantage of the labeled data from the target domain using active learning techniques. To this end, [Chan and Ng, 2007] proposed a method where active learning is used for word sense disambiguation in a transfer learning setting. Their active learning setting is pool-based whereas [Rai *et al.*, 2010], propose a similar method but in a streaming (online) setting, as a result there is not the requirement of an initial pool of labeled target domain. Nevertheless, both methods are applicable when labeled data exist also in the target domain.

Transductive transfer learning Many approaches of this type are motivated by importance sampling. Their motivation is to add weights to instances, using the probability density ratio (i.e. the difference of the source and target distributions). For example, kernel-mean matching (KMM) algorithm is proposed in [Huang *et al.*, 2007], to learn directly the density ratio, by matching the means between the source and the target domain data in a reproducing-kernel Hilbert space (RKHS). In the same vein, an algorithm known as Kullback-Leibler Importance Estimation Procedure (KLIEP) is proposed in [Sugiyama *et al.*, 2007], in order to estimate the difference of the source and target distributions directly, based on the minimization of the Kullback-Leibler divergence. In [Jiang and Zhai, 2007], the proposed approach uses instance weighting, by adding instance-dependent weights to the loss function. Another family of transductive transfer learning approaches are the feature-representation-transfer ones. For example, a structural correspondence learning (SCL) algorithm is proposed in [Blitzer *et al.*, 2006], to make use of the unlabeled data from the target domain and extract some relevant features that may reduce the difference between the domains. The effect of representation change for domain adaptation is also analyzed in [Ben-David *et al.*, 2007]. Also, a coclustering based approach is presented in [Dai *et al.*, 2007a], aiming to propagate the class information from the target to source domain, by identifying word clusters shared among the two domains. Transfer learning via dimensionality reduction was proposed in [Pan *et al.*, 2008]. They exploited the Maximum Mean Discrepancy Embedding (MMDE) method, originally designed for dimensionality reduction, to learn a low-dimensional space that reduces the difference of distributions between different domains. However, MMDE has been proved computationally expensive. Thus, in [Pan *et al.*, 2009], Transfer Component Analysis (TCA) is proposed, which uses an efficient feature extraction algorithm. In [Daume and Marcu, 2006], an approach based on a mixture model is presented. Their key idea is to assume that source domain data are drawn from a mixture of two distributions: a truly "in-domain" distribution and a "general domain" one. Similarly, the target domain data is treated as if drawn from a mixture of "out-of-domain" distribution and the "general domain" distribution, as the source domain data. Approaches that target natural languages try to exploit external knowledge sources or various kinds of linguistic information. In [Gabrilovich and Markovitch, 2005] an approach that extracts new features by exploiting world knowledge is presented. World knowledge is represented through publicly available ontologies, such as the Open Directory Project (ODP), where features from the source domain are mapped to appropriate ontology concepts, and "is-a" relations are exploited in order to acquire new features that augment the original feature set. Finally, the most appropriate features are selected through a feature selection phase. The work presented in [Zhang and Shakya, 2009] exploits feature correlation in order to group features into correlated groups. For example, words like "orange", "lemon", "apple" and "pear" may often appear together in documents: aggregating them into a new correlated group "fruits", creates a new feature. If enough evidence exists in a document from the target domain (i.e.some of the features of the correlated group appear in the document), the feature that corresponds to the correlated group may help the task T_T in the target domain.

Transfer learning in sentiment analysis Transfer learning approaches have already been applied for sentiment analysis, under different settings. In [Glorot *et al.*, 2011], the aim is to create a model from a set of domains containing labeled and unlabeled data, while applying the resulting model to any other domain. To this end the authors apply domain adaptation on features extracted through a deep-learning method application. They work on an Amazon review dataset [Blitzer *et al.*, 2007b], extracting "deep" features from the whole set of instances across domains. Then, these features are used to train a linear SVM on the labeled instances (source domain). The classification is then performed in the deep feature space, using the trained SVM. They illustrate good performance, in terms of averaged transfer generalization error, across 4 domains.

In [You *et al.*, 2015], a system draws on millions of Flickr images (from SentiBank) to create a weak-learner-labeled dataset, then trains a Progressive Convolutional Neural Network (PCNN) to perform Twitter image sentiment analysis. Using training sets of about 400K (random) instances, they achieve a performance of about 78% (in F-measure and accuracy) over a set of 44K Flickr image instances. Before the transfer to Twitter, the performance is measured on a set of about 1200 manually labeled instances with an achieved level of performance of 75% to 82% F-measure (measured on different subsets of the data, ranked by level of human agreement). After the transfer, where labeled Twitter instances are used in a cross-validation process, the transferred (i.e. finetuned) PCNN model increases the performance by 1 or 2 percentile units.

In [Calais Guerra *et al.*, 2011], user bias when expressing sentiment is detected. This bias is then transferred to textual features to aid real-time sentiment classification. In their transfer learning case, there exists a target task (real-time sentiment analysis, with no training data and presence of concept drift) and a source task (opinion holder bias prediction). The labels of the source task are mapped to labels in the target task. The user bias is extracted based on relational learning on social media endorsements among users. The user bias (as a function over terms) is then applied to determine the polarity of terms. This term polarity is then used to describe the polarity of tweets, showing very stable and promising performance (around 87% F-measure over 400K documents on politics and 50K documents on soccer) as compared to an SVM classifier that is trained on textual-only, Twitter data.

In [Yoshida *et al.*, 2011] the authors propose a Bayesian model, which infers domain (in)dependence of words and word polarity per domain to solve a multi-domain transfer learning problem on sentiment analysis. In their evaluation they use 17 domains from the Amazon review dataset [Blitzer *et al.*, 2007b] over 10K documents. They select 3 random domains as target domains and vary the source domains (from 1 to 14) to evaluate the F-measure at the target. They show that including the domain, as well as the domain dependence of a term improves the performance and stability of the system (best F-measure achieved around 67% in a cross-validation setting).

Finally, in [Li *et al.*, 2013] the authors apply active learning to select unlabeled instances from the target domain, so that a source and a target classifier are trained. The unlabeled instances are assigned labels by Label Propagation and enrich the two classifiers, which are combined (multiplication of individual class probabilities for a given instance) to offer the final classification of instances. The experiments are conducted using 4 domains of the multi-domain Amazon review dataset. The results show that the proposed active learning method, combined with the transfer approach perform equally well to established methods, reaching a level of performance of about 80% accuracy.

The proposed approach differs from the above, as it tries to capture the difference of the domains, by calculating the difference of word distributions. In particular we use KLIEP, a well-known approach to calculate this difference. The latter is used as instance weights and are incorporated in PLSA algorithm. The intuition of the proposed approach is that words (and in extension documents) have varied importance between domains. The weights we propose, try to capture the importance of each instance, by comparing the word distribution across the domains. In the next section, the proposed approach is presented in details.

3 Proposed approach: KLIEP-PLSA

We introduce a transfer learning classifier, which, tries to capture the domain difference in the word distribution level. To this end, in the section, we present the Probabilistic Latent Semantic Analysis (PLSA) [Hofmann, 2001] model, in which we incorporate instance weights, based on the potential importance of each training instance in the test domain. We first present the KLIEP [Sugiyama *et al.*, 2007] approach, which is used to calculate the weights. Then we present the PLSA model. The extended PLSA model which incorporate these weights is then described.

3.1 Kullback-Leibler Importance Estimation Procedure (KLIEP)

A situation where the input distribution $P(x)$ is different in the training and test phases but the conditional distribution of output values, $P(y|x)$, remains unchanged is called covariate shift [Shimodaira, 2000]. The influence of covariate shift could be alleviated by weighting the log likelihood terms according to the importance [Shimodaira, 2000]: $w(x) = \frac{P_T(x)}{P_S(x)}$, where $P_T(x)$ and $P_S(x)$ are target and source input densities. Since the importance is usually unknown, the key issue of covariate shift adaptation is how to accurately estimate the importance.

KLIEP is an approach proposed by [Sugiyama *et al.*, 2007] for directly estimating the ratio of two density functions without going through density estimation. KLIEP expresses the weights as a linear model and determines the parameters of the model so that the Kullback-Leibler divergence from the target distribution to the source distribution is minimized. The optimization of KLIEP is also a convex optimization. Therefore a global solution can be obtained.

In order to determine a weight $w(x)$ of a data point x without distribution estimation, the weight is modeled in KLIEP as a linear model. That is:

$$\hat{w}(x) = \sum_{l=1}^{b} \alpha_l \phi_l(x) \tag{1}$$

where $\{\alpha_l\}_{l=1}^{b}$ is a parameter to be learned and $\{\phi_l(x)\}_{l=1}^{b}$ is a basis function such that

$$\phi_l(x) \geq 0 \text{ for all } x \in D \text{ and for } l = 1, 2, \dots, b \tag{2}$$

We use Gaussian Kernel as $\phi_l(x)$, i.e. $K(x, x') = \exp\left(-\frac{||x-x'||^2}{2\sigma^2}\right)$.

Since target distribution can be approximated from $w(x)$ and source distribution, the goal is to find an optimal $\hat{w}(x)$. The optimal weights $\hat{w}(x)$ are obtained by minimizing Kullback-Leibler divergence between $P_T(x)$ and $\hat{P}_T(x)$, where $\hat{P}_T(x)$ is an empirical distribution of $P_T(x)$ given as

$$\hat{P}_T(x) = \hat{w}(x) P_S(x) \tag{3}$$

We determine the parameters $\{a_l\}_{l=1}^{b}$ in the model 1,so that the Kullback-Leibler divergence between $P_T(x)$ and $\hat{P}_T(x)$ is minimized:

$$KL[P_T(x)||\hat{P}_T(x)] = \int_D P_T(x) \log \frac{P_T(x)}{\hat{w}(x) P_S(x)} dx$$

$$= \underbrace{\int_D P_T(x) \log \frac{P_T(x)}{P_S(x)} dx}_{constant} - \int_D P_T(x) \log \hat{w}(x) dx$$

(it is constant, as it is independent of $\{a_l\}_{l=1}^{b}$).

We denote with J the second term:

$$\begin{aligned} J &:= \int_D P_T(x) \log \hat{w}(x) dx \\ &\approx \frac{1}{n_T} \sum_{j=1}^{n_T} \log \hat{w}(x_j^T) = \frac{1}{n_T} \sum_{j=1}^{n_T} \log \left(\sum_{l=1}^{b} \alpha_l \phi_l(x_j^T) \right) \end{aligned}$$

Now, the optimization criterion is summarized as follows:

$$\max \left[\sum_{j=1}^{n_T} \log \left(\sum_{l=1}^{b} \alpha_l \phi_l(x_j^T) \right) \right] \tag{4}$$

subject to $\sum_{i=1}^{n} \sum_{l=1}^{b} \alpha_l \phi_l(x_i^S) = n$ and $\alpha_1, \alpha_2, \dots, \alpha_b \geq 0$

This is a convex optimization problem and the global solution can be obtained by simply performing gradient ascent and feasibility satisfaction iteratively. For more details in the implementation of the model, please refer to [Sugiyama *et al.*, 2007].

3.2 Probabilistic Latent Semantic Analysis

PLSA is a probabilistic model which characterizes each word in a document as a sample from a mixture model, where mixture components are conditionally-independent multinomial distributions. It has been proposed as a probabilistic version of the Latent Semantic Analysis (LSA) method [Dempster *et al.*, 1977]. This model associates an unobserved latent variable (called aspect, topic or component) $k \in \{k_1, ..., k_K\}$ to each observation corresponding to the occurrence of a word $f \in \mathcal{F}$ within a document $x \in \mathcal{X}$. One component or topic can coincide with one class or, in another setting, a class may be associated with more than one component. Although originally proposed in an unsupervised setting, this latent variable model is easily extended to classification with the following underlying generation process:

- Pick an example x with probability $P(x)$,
- Choose a latent variable k according to its conditional probability $P(k \mid x)$
- Generate a feature f with probability $P(f \mid k)$
- Generate the example's class y according to the probability $P(y \mid k)$.

The probability $P(y \mid k)$ is fixed, by forcing to zero the component k that do not belong to a certain class y, i.e. $P(y|k) = \begin{cases} 1, \text{if } k \in y \\ 0, \text{otherwise} \end{cases}$ (as we know a priori how many components per class we have).

Hence, the model parameters are
$\Xi = \{P(k \mid x), P(f \mid k) : k \in K, x \in \mathcal{X}, f \in \mathcal{F}\}$

The generation of a feature f within an example x can then be translated by the following joint probability model:

$$P(f, x) = P(x) \sum_{k \in K} P(w \mid k) P(k \mid x) \tag{5}$$

So, the log-likelihood of the model can be estimated as:

$$\mathcal{L} = \sum_{f \in \mathcal{F}} \sum_{x \in \mathcal{X}} n(x, f) \log P(x, f) \tag{6}$$

where $n(f, x)$ denotes the frequency of the word f in instance x.

Figure 1 shows the graphical model for PLSA. The parameters of $P(f \mid k)$, $P(x \mid k)$, and $P(k)$ over all f, x, k are obtained by EM estimation of the maximum likelihood.

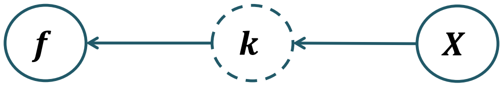


Figure 1: Graphical model representation of PLSA. Latent variables are indicated by dotted circles.

3.3 Proposed approach

In this work, we combine the weights calculated by KLIEP with PLSA algorithm. In order to *incorporate the weights* as calculated by KLIEP, and as a result, take into account the difference between the word distribution across domains, we replace in equation 5, the $P_T(x)$ by:

$$P_T(x) = w(x)P_S(x) \tag{7}$$

where $P_T(x)$ and $P_S(x)$ correspond to the distributions of the data in the test and train domain respectively.

In algorithm (1) the model is described. For the initialization of the model ($\Xi^{(0)}$), we force to zero the $P(k \mid x)$ for an example x which does not belong to a particular topic k (that is the labeled training examples), and give random values to the rest, under the constraint to sum up to 1 (i.e. the unlabeled test examples). The $P(f \mid k)$ is initialized by giving random values for all f and k. In addition, we calculate the training instance weights using KLIEP.

After the training of the model, we classify the examples of the test set with the maximum posterior probability using chain rule:

$$P(y|x) \propto \sum_k P(k|x)P(y|k)$$

We choose as label for each example, the one with the highest probability.

At the initialization of the model ($\Xi^{(0)}$), we use the constraints introduced above for $P(x \mid z)$. The $P(f \mid k)$ is initialized with random values for all f and k.

Once the model is trained, we then classify the documents in one of the classes using chain rule:

$$P(z \mid x) \propto P(x \mid z)P(z) = P(x \mid z)\sum_k P(k, z) \tag{8}$$

We choose as label for each document, the one with the highest probability, taking into account that there is a one-to-one matching between document topics k and classes:

$$argmax_k P(x \mid k) \tag{9}$$

In addition, as our model is a generative one, we can run KLIEP-PLSA for new documents (x_{new}) from the target domain, using the calculated model (i.e.$P(f|k)$), in order to learn the $P(x_{new}|k)$.

4 Evaluation

In order to evaluate the algorithm proposed in the previous section, we performed experiments on the customer review dataset presented in [Blitzer *et al.*, 2007a; Dasgupta and Ng, 2010]. The dataset contains product reviews from four different domains: Book (B), DVD (D), Electronics (E) and Kitchen (K) appliances, each of which contains 1000 positive and 1000 negative labeled reviews from Amazon. The reviews are represented as bag-of-words. The goal is to classify each review as positive or negative. We examine the performance of the system on all the possible pairs of (source, target) domains: (book, electronics), (book, DVD), (electronics, DVD),(electronics, book), (DVD, book), (DVD, electronics), using the pre-defined partitions.

Algorithm 1 Training of KLIEP-PLSA

```
Input:
  • Data from source and target domains $\mathcal{X}_S$ and $\mathcal{X}_T$,
  • Random initial model parameters $\Xi^{(0)}$.
  • Calculation of instance weights using KLIEP
  • $j \leftarrow 0$
repeat
  • E-step: Estimate the latent class posteriors:
    $\forall x \in \mathcal{X}, \forall f \in \mathcal{F}, \forall k \in K, \forall z \in Z$
      $P^{(j)}(k|f,x) = \frac{P^{(j)}(k \mid x)P^{(j)}(f \mid k)}{\sum_{k' \in K} P^{(j)}(k' \mid x)P^{(j)}(f \mid k')}$
  • M-step: Estimate the new model parameters $\Xi^{(j+1)}$
    by maximizing the complete-data log-likelihood:
      $P^{(j+1)}(f \mid k) \propto \sum_x n(f,x)P^{(j)}(k|f,x)$
      $P^{(j+1)}(k \mid x) \propto \sum_f n(f,x)P^{(j)}(k|f,x)$
  • $j \leftarrow j+1$
until convergence of the complete-data log-likelihood
Output: A generative classifier with parameters $\Xi^{(j)}$
```

4.1 Evaluation measures

To measure the performance of the method, we use the F1-measure, which is the harmonic mean of precision and recall.

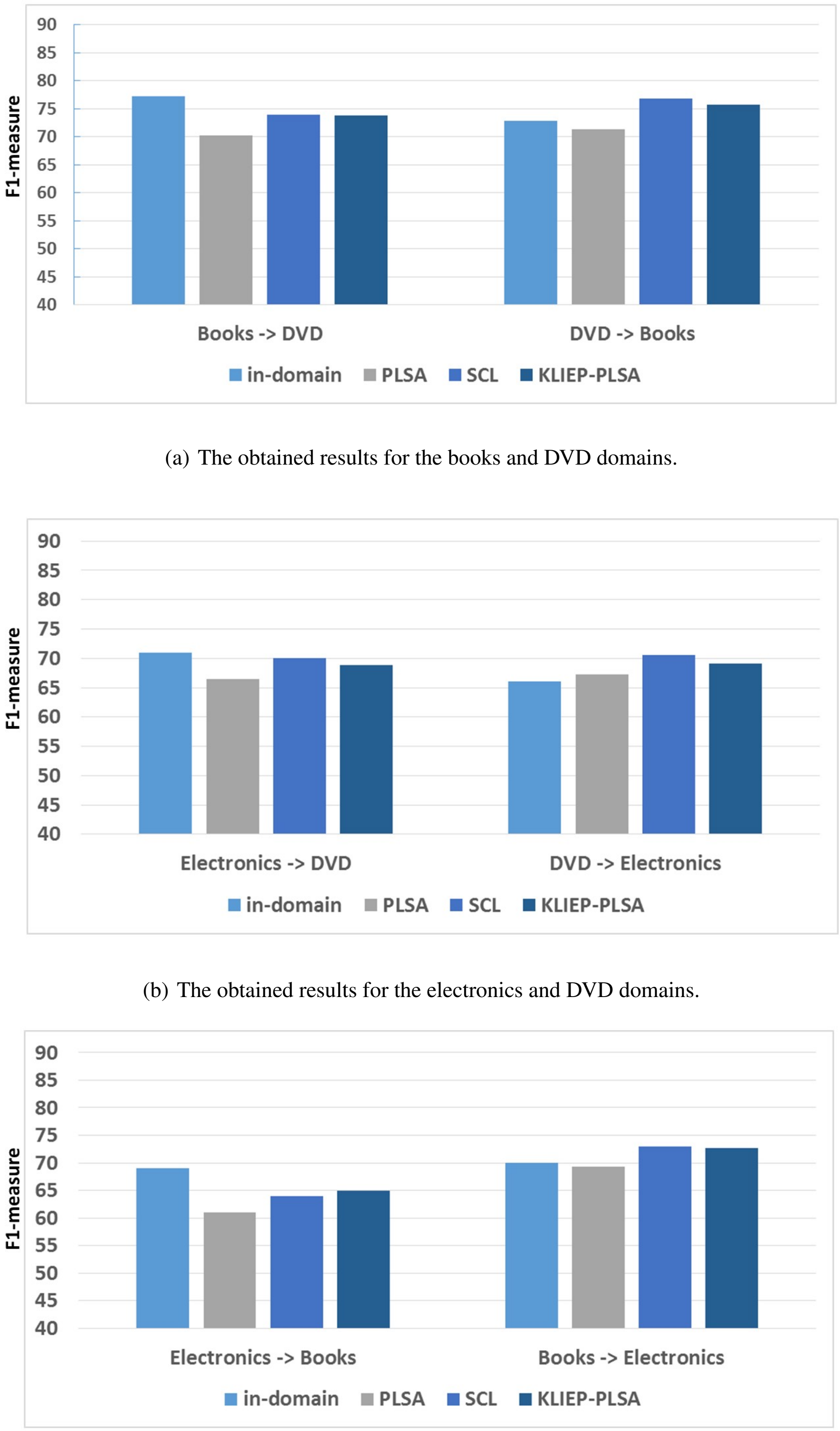
$$F_1 = 2 \times \frac{P \times R}{P + R} \tag{10}$$

where

$$P = \frac{TP}{TP + TN}, R = \frac{TP}{TP + FN},$$

TP is the number of "true positives" (instances that were labeled positive correctly), TN is the number of "true negatives" (instances that were labeled negative correctly), FP, FN is the number of "false positives" and "false negatives" (instances that were incorrectly labeled as positive or negative) correspondingly.

We note that, since the problem is a binary classification ("positive" and "negative" classes), calculating F1-measure over the positive class is enough to illustrate performance over both classes.



(a) The obtained results for the books and DVD domains.

(b) The obtained results for the electronics and DVD domains.

(c) The obtained results for the electronics and books domains.

Figure 2: The performance of the KLIEP-PLSA, compared with the other approaches for the different source-target pairs.

Table 1: The obtained results in the different pairs of domains. The presented scores are the F1-measure. The values in bold are the ones with significant difference.

	Method			
Source - Target Domains	**in-domain**	**PLSA**	**SCL[Blitzer *et al.*, 2007a]**	**KLIEP-PLSA**
Books → DVD	**77.2**	70.2	74	73.8
DVD → Books	72.8	71.3	**76.8**	75.7
Electronics → DVD	**71**	66.4	70	68.9
DVD → Electronics	66	67.2	70.5	69.1
Electronics → Books	69	61	64	65
Books → Electronics	70	69.3	73	72.7

4.2 Results

We compared the performance of the model on the above datasets in all different combinations of source and target domains. We performed 10 runs for each combination and we calculated the average F-score. As we initialize some the training parameters of PLSA at random, we wanted to get representative performance for multiple random initializations. In order to evaluate the significance of the observed differences in performance, we performed a t-test at the 5% significance level.

We compared the proposed approach with two approaches: The simple PLSA, in order to evaluate the combination with KLIEP, and with Structural Correspondence Learning (SCL) algorithm [Blitzer *et al.*, 2007a], a well-known transfer learning approach, which has been evaluated in the task of sentiment analysis across domains. As a reference, we present the in-domain results, as presented in [Li *et al.*, 2013]. The latter is based on the maximum entropy (ME) classifier trained on the target domain.

In table 1 and figure 2, the obtained results are presented. As we can notice, the use of weights improves the performance the PLSA, which means that the density ratio can actually capture the difference in the word distribution across domains. In addition, the proposed approach, even though it does not outperform the SCL results, it manages to achieve comparable performance. It has to be noticed that no significant statistical difference has occurred on the obtained results between SCL and KLIEP-PLSA, but in only one of the pairs, namely DVD-Books (table 1). One of the advantages of the proposed approach is that is language agnostic, as it is based on the statistical distribution of words/tokens, and not on any language specific information.

5 Conclusions

In this paper, we presented KLIEP-PLSA, a new transfer learning approach, based on PLSA. The motivation for this work is to use transfer learning, for the task of sentiment analysis. The evaluation shows that the proposed approach managed to capture in most cases the difference between domains, and in particular word distribution across domains. Our approach outperforms the simple PLSA method, and achieves comparable results with SCL approach.

Our immediate next target is to further investigate KLIEP-PLSA, in order to better understand how the density ratio can help us capture the difference between domains. In addition, more extensive experiments should be performed in order to further evaluate the approach. Another future direction we consider is the use of multiple domains for training, in order to evaluate the impact of the amount of training data in respect to accuracy of our method.

Acknowledgments

The research leading to these results has received funding from the European Regional Development Fund of the European Union and from national funds in the context of the research projects 1) 'SentIMAGi - Brand monitoring and reputation management via multi-modal sentiment analysis' (ISR_ 2935) under the Regional Operational Programme Attica (Priority Axis 3 "Improving competitiveness, innovation and digital convergence") of the 'Bilateral R& D Cooperation between Greece and Israel 2013-2015' of the Action of national scope 'Bilateral , Multilateral and Regional R& D Cooperation' and 2) 'Construction of repository and interface with online linguistic services' (MIS 441451), of the action "CLARIN ATTICA Support and development of greek partners for the participation in the Consortia of the European Research Infrastructures in the context of the ESFRI/2006" under the Regional Operational Programme "Attica" (Priority Axis 3 "Improving competitiveness, innovation and digital convergence".

References

[Ben-David *et al.*, 2007] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira. Analysis of representations for domain adaptation. In *In Advances in Neural Information Processing Systems (NIPS)*. MIT Press, 2007.

[Blitzer *et al.*, 2006] J. Blitzer, R. McDonald, and F. Pereira. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, 2006.

[Blitzer *et al.*, 2007a] John Blitzer, Mark Dredze, and Fernando Pereira. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. *ACL*, 7:440–447, 2007.

[Blitzer *et al.*, 2007b] John Blitzer, Mark Dredze, Fernando Pereira, et al. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *ACL*, volume 7, page 440447, 2007.

[Calais Guerra *et al.*, 2011] Pedro Henrique Calais Guerra, Adriano Veloso, Wagner Meira Jr, and Virglio Almeida. From bias to opinion: a transfer-learning approach to real-time sentiment analysis. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, page 150158. ACM, 2011.

[Chan and Ng, 2007] Y. Chan and H. Ng. Domain adaptation with active learning for word sense disambiguation. In *Proceedings of the 45th Annual Meeting of the ACL*, June 2007.

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

[Dai *et al.*, 2007b] W. Dai, Q. Yang, G. Xue, and Y. Yu. Boosting for transfer learning. In *Proceedings of the 24th international conference on Machine learning*, ICML '07, 2007.

[Dasgupta and Ng, 2010] Sajib Dasgupta and Vincent Ng. Mining clustering dimensions. In *Proceedings of the 27th International Conference on Machine Learning*, pages 263–270, 2010.

[Daume and Marcu, 2006] H. Daume, III and D. Marcu. Domain adaptation for statistical classifiers. *Journal of Artificial Intelligence Research*, 26(1):101–126, 2006.

[Daume *et al.*, 2010] H. Daume, III, A. Kumar, and A. Saha. Frustratingly easy semi-supervised domain adaptation. In *In Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*, 2010.

[Dempster *et al.*, 1977] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Statistical Society, Series B*, 39(1), 1977.

[Gabrilovich and Markovitch, 2005] Evgeniy Gabrilovich and Shaul Markovitch. Feature generation for text categorization using world knowledge. In *Proceedings of the 19th international joint conference on Artificial intelligence*, IJCAI'05, pages 1048–1053, San Francisco, CA, USA, 2005.

[Glorot *et al.*, 2011] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, page 513520, 2011.

[Hofmann, 2001] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Mach. Learn.*, 42(1-2):177–196, 2001.

[Huang *et al.*, 2007] J. Huang, A. Smola, A. Gretton, Ka. Borgwardt, and B. Schölkopf. Correcting sample selection bias by unlabeled data. In *Advances in Neural Information Processing Systems 19*, 2007.

[III *et al.*, 2010] Hal Daumé III, Tejaswini Deoskar, David McClosky, Barbara Plank, and Jörg Tiedemann, editors. *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*. Association for Computational Linguistics, Uppsala, Sweden, July 2010.

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

[Lee *et al.*, 2007] S. Lee, V. Chatalbashev, D. Vickrey, and D. Koller. Learning a meta-level prior for feature relevance from multiple related tasks. In *Proceedings of the 24th international conference on Machine learning*, ICML '07, 2007.

[Li *et al.*, 2013] Shoushan Li, Yunxia Xue, Zhongqing Wang, and Guodong Zhou. Active learning for cross-domain sentiment classification. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, page 21272133. AAAI Press, 2013.

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

[Pan *et al.*, 2008] S. J. Pan, J. T. Kwok, and Q. Yang. Transfer learning via dimensionality reduction. In *Proceedings of the 23rd national conference on Artificial intelligence - Volume 2*, AAAI'08, 2008.

[Pan *et al.*, 2009] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang. Domain adaptation via transfer component analysis. In *Proceedings of the 21st IJCAI*, 2009.

[Rai *et al.*, 2010] P. Rai, A. Saha, H. Daumé, III, and S. Venkatasubramanian. Domain adaptation meets active learning. In *Proceedings of the NAACL HLT 2010 Workshop on Active Learning for Natural Language Processing*, ALNLP '10, 2010.

[Shimodaira, 2000] Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, October 2000.

[Sugiyama *et al.*, 2007] M. Sugiyama, S. Nakajima, H. Kashima, P. von Bnau, and M. Kawanabe. Direct importance estimation with model selection and its application to covariate shift adaptation. In *NIPS*, 2007.

[Yoshida *et al.*, 2011] Yasuhisa Yoshida, Tsutomu Hirao, Tomoharu Iwata, Masaaki Nagata, and Yuji Matsumoto. Transfer learning for multiple-domain sentiment analysisidentifying domain dependent/independent word polarity. In *Twenty-Fifth AAAI Conference on Artificial Intelligence*, 2011.

[You *et al.*, 2015] Quanzeng You, Jiebo Luo, Hailin Jin, and Jianchao Yang. Robust image sentiment analysis using progressively trained and domain transferred deep networks. In *The Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI)*, 2015.

[Zhang and Shakya, 2009] Jian Zhang and Shobhit S. Shakya. Knowledge transfer for feature generation in document classification. In *Proceedings of the 2009 ICMLA*, pages 255–260, Washington, DC, USA, 2009. IEEE Computer Society.