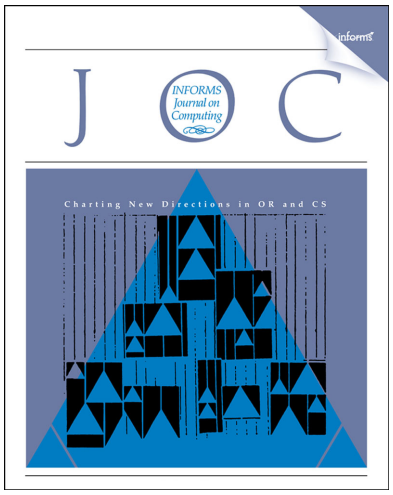

INFORMS Journal on Computing

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Domain Adaptation for Sentiment Classification in Light of Multiple Sources

Fang Fang, Kaushik Dutta, Anindya Datta

To cite this article:
Fang Fang, Kaushik Dutta, Anindya Datta (2014) Domain Adaptation for Sentiment Classification in Light of Multiple Sources. INFORMS Journal on Computing 26(3):586-598. https://doi.org/10.1287/ijoc.2013.0585

INFORMS Journal on Computing
Vol. 26, No. 3, Summer 2014, pp. 586–598
ISSN 1091-9856 (print) | ISSN 1526-5528 (online)

http://dx.doi.org/10.1287/ijoc.2013.0585

Domain Adaptation for Sentiment Classification in Light of Multiple Sources

Fang Fang, Kaushik Dutta, Anindya Datta
Department of Information Systems, National University of Singapore, Singapore 117418
{fangfang@comp.nus.edu.sg, duttak@nus.edu.sg, datta@comp.nus.edu.sg}

Sentiment classification is one of the most extensively studied problems in sentiment analysis, and supervised learning methods, which require labeled data for training, have been proven quite effective. However, supervised methods assume that the training domain and the testing domain share the same distribution; otherwise, accuracy drops dramatically. Although this does not pose problems when training data are readily available, in some circumstances, labeled data is quite expensive to acquire. For instance, if we want to detect sentiment from Tweets or Facebook comments, the only way to acquire is to manually label it, and this is prohibitively burdensome and time-consuming. In this paper, we propose a hybrid approach that integrates the sentiment information from source-domain labeled data and a set of preselected sentiment words to solve this problem. The experimental results suggest that our method statistically outperforms the state of the art and even, in some cases, surpasses the in-domain gold standard.

Keywords: sentiment analysis; business intelligence; domain adaptation
History: Accepted by Alexander Tuzhilin, (former) Area Editor for Knowledge and Data Management; received October 2012; revised July 2013; accepted October 2013. Published online in *Articles in Advance* May 5, 2014.

1. Introduction

With the explosion of blogs, social networks, reviews, ratings as well as other user-generated texts, *sentiment analysis*, which aims to detect the underlying sentiments embedded in those texts, has attracted much research interest recently. Such sentiments are useful to various constituencies: (a) Consumers can use sentiment analysis to research products or services before making a purchase. (b) Marketers can use this to research public opinion regarding their company and products, or to analyze customer satisfaction. Finally, (c) organizations can also use this to gather critical feedback about problems in newly released products.

One of the tasks of sentiment analysis is to determine the overall sentiment orientation of a piece of text. This problem has been widely investigated and *supervised learning methods*, which require labeled data for training, have been proven quite effective. However, supervised methods assume that the training data domain and the testing data domain share exactly the same distribution; i.e., (a) texts in both data sets are represented in the same feature space, and (b) features, or words, follow the same distributions in both data sets. The first assumption requires that a similar set of words are used in both domains, whereas the second assumption demands that the occurrence probability of a word is identical in training and testing domains. If these assumptions do not hold, accuracy drops dramatically (about 10% according to our experiment results). These assumptions do not pose problems when performing sentiment analysis in domains where training data are readily available. An example of such a domain is movie reviews. Each review is typically accompanied by a numerical rating, allowing easy assignment of sentiment to the review. In nearly all previous work, reviews rated 1 and 2 are considered as negative and those rated 4 and 5 are treated as positive. However, in circumstances where user-assigned ratings are not available, labeled data is quite expensive to acquire. For instance, if we want to detect sentiment from Tweets or comments in Facebook, the only way to get labeled data is to manually label it, and this is prohibitively burdensome and time-consuming. Yet, sentiment mining is pervasive enough so that its application is useful in many domains, such as Tweets and Facebook comments, where labeled data are not available.

This is the problem addressed in this paper. We want to determine the sentiment orientation of a piece of text when *in-domain* labeled data are not available. A number of methods have been proposed in the literature most of which rely on the idea of applying labeled data from a "source" domain to perform sentiment classification on data in a different "target" domain through domain-independent features called *pivot* features. Following is an illustrative example.

Suppose we are adapting from "computers" domain to "cell phones" domain. While many of the features of a good cell phone review are the same as a computer review, such as "excellent" and "awful," many words are totally new, like "reception." In addition, many features that are useful for computers, for instance "dual-core," are not useful for cell phones. The intuition is that even though the phrase "good-quality reception" and "fast dual-core" are completely distinct for each domain, they both have high correlation with "excellent" and low correlation with "awful" on unlabeled data. As a result, we can tentatively align them (Blitzer et al. 2007). After learning a classifier for computer reviews, when we see a cell-phone feature like "good-quality reception," we know it should behave in a roughly similar manner to "fast dual-core."

The main drawback of these methods is that the performance is largely dependent on the selection of pivot features. Ideally, pivot features would act similarly in both target and source domains toward sentiment. The problem is that we do not know the sentiment of the data in the target domain, making it extremely hard to select those pivot features accurately.

In this paper, we propose a hybrid approach that integrates the sentiment information from labeled data of multiple source domains and a set of preselected sentiment words for sentimental domain adaptation, i.e., *cross-domain* sentiment classification. To solve the aforementioned limitation caused by difficulty of pivot feature selection, we tackle this task by mapping the data into a latent space to learn an abstract representation of the text. The assumption we make is that texts with the same sentiment label would have similar abstract representations, even though their text representations differ. For instance, in the previous example, the phrase "good-quality reception" and "fast dual-core" are completely distinct for each domain; however, in the latent space, they might correspond to the same feature. This idea has been used in Titov (2011) and Glorot et al. (2011); however, as we will discuss later, our method is distinct enough from them. Furthermore, in addition to the use of *out-domain* data, we also utilize sentiment information from preselected opinionated words. We believe these words could provide certain helpful sentiment information in our classification context. Finally, we train our classifiers over the new hybrid representations. The experimental results suggest that our method statistically outperforms the state of the art and even surpasses the *in-domain* method in some cases.

The rest of the paper is organized as follows: We first review related work in literature. Then we provide the intuition and overview of our method followed by an elaboration of our proposed method. Thereafter, we evaluate our method on a benchmark data set. Finally, we conclude our paper with a discussion of this study.

2. Related Work

In this section, we review related work on in-domain sentiment classification, cross-domain sentiment classification, and other sentiment analysis tasks.

2.1. In-Domain Sentiment Classification

One of the most thoroughly studied problems in sentiment analysis is the in-domain sentiment classification, which refers to the process of determining the overall tonality of a piece of text and classifying it into several sentiment classes. Two main research directions have been explored, i.e., document-level sentiment classification and sentence-level sentiment classification.

In document-level classification, documents are assumed to be opinionated and all documents are classified as either positive or negative (Liu 2010). This problem can be addressed as either a supervised learning problem or an unsupervised classification problem. Many of the existing research using the supervised machine learning approach has used product reviews as target documents. Training and testing data are very convenient to collect for these documents since each review already has a reviewer-assigned rating, typically one to five stars. One representative work would be Pang and Lee (2008). They employed multiple approaches to the sentiment classification problem and concluded that machine learning methods definitively outperform others.

Due to opinion words being the dominating indicators for sentiment classification, it is quite natural to use unsupervised learning based on such words. This kind of method has not been studied so much because of its relatively inferior performance compared with supervised methods. The simplest method is to determine the sentiment of a document based on the occurrences of positive and negative words. A review could be classified as positive if there are more positive words and categorized as negative otherwise. One representative example of more sophisticated work is Turney (2002). They performed classification based on certain fixed syntactic phrases that are likely to be used to express opinion. They first identified phrases with positive semantic orientation and phrases with negative semantic orientation. The semantic orientation of a phrase was calculated as the mutual information between the given phrase and the word "excellent" minus the mutual information between the given phrase and the word "poor." A review was classified as positive if the average semantic orientation of its phrases is positive and categorized as negative otherwise.

In sentence-level classification, sentences are first classified as subjective or objective. Then subjective sentences are further classified into positive or negative (Liu 2010). Traditional supervised learning methods have been applied here. Representative examples include Wiebet and Bruce (1999), which used a naïve Bayesian classifier for subjectivity classification. Other learning algorithms are also used in subsequent research (Hatzivassiloglou and Wiebe 2000, Riloff and Wiebe 2003). One of the bottlenecks for this task is the lack of training example. A bootstrapping approach to automatically label training data was proposed in Riloff and Wiebe (2003) to solve this problem.

2.2. Cross-Domain Sentiment Classification

Most sentiment classification methods assume that training data and testing data share exactly the same distribution. The assumption can be interpreted from two perspectives: (a) documents in both training domain and testing domain are represented using the same set of words; (b) words follow the same distribution. The first perspective necessitates that the same set of words are used in both the training domain and the testing domain while the second part obliges that the probability of a word occurring in the training domain equals that in the testing domain. If these two assumptions are not met, accuracy of the classifier drops dramatically. A number of solutions have been proposed to solve this problem and all of them utilize labeled data from other domains, or source domains. Intuition in most existing research is to map features between the target domain and the source domain making use of domain-independent features known as pivot features. An illustrative example is given in the introduction. Two kinds of pivot features were explored in literature: words (Blitzer et al. 2007, Bollegala et al. 2011, Pan et al. 2010) and topics (He et al. 2011, Liu and Zhao 2009). We discuss them in turn next.

Blitzer et al. (2007) started the line of research on cross-domain sentiment classification. They selected words as pivot features according to their common frequency and mutual information with the source labels, and then applied a structural correspondence learning (SCL) algorithm to obtain k new real-valued features. Finally, they augmented the original feature with the k new real-valued features in both the source domain and the target domain, and performed classification over the new feature space. Pan et al. (2010) also proposed a similar method. They selected words with low mutual information between words and domains as pivot features, and then ran a spectral feature alignment (SFA) algorithm to align domain-specific words. The classification was performed over the augmented feature space. Bollegala et al. (2011) also used words as pivot features but in a different manner. Instead of selecting a small set of domain-independent features, they treated all features as pivot features. Based on pointwise mutual information, relatedness between any two words was calculated. Then, they expanded the feature representation of a document with those words that are highly related with words in the document and trained classifiers over the new feature space. So far this is the only work that has used multiple source domains simultaneously. Multiple source domains can also be used simultaneously in our approach but in a different manner. For example, (a) we use a latent space model to learn latent representations; (b) we only rely on the newly learnt features, and original word features are discarded in our approach; and (c) we use sentiment information from preselected opinionated words in our method.

With the success of the topic model, researchers also attempted to use topics as pivot features. Liu and Zhao (2009) observed that customers often use different words to comment on similar topics in different domains, and therefore, these common topics can be used as the bridge to link different domain-specific features. They proposed a topic model named transfer-PLSA to extract the topic knowledge across different domains. Through these common topics, the features in the source domain were mapped to the target domain features, so that the domain-specific knowledge could be transferred across different domains. He et al. (2011) also proposed a similar method using the joint sentiment-topic (JST) model which incorporates word polarity priors through modifying the topic-word Dirichlet priors.

All work discussed so far used pivot features, and their experimental results suggest that classification accuracies have been improved. However, pivot features have limitations. Ideally, pivot features, or domain-independent features, would act exactly the same way with respect to sentiment labels in both domains. However, it is hard to measure since we do not have labeled data in target domains and performance would largely depend on selection of pivot features. To break this limitation, latent space models were introduced for cross-domain sentiment classification. Titov (2011) used a the harmonium model of Smolensky (1986) with a single layer of binary latent variables to cluster features in both domains and ensure that at least some of the latent variables are predictive of the label on the source domain. Such a model can be regarded as composed of two parts: a mapping from an initial (normally, word-based) representation to a new shared distributed representation, and a classifier in this representation. They combined their model with the baseline out-domain model using the product-of-experts combination (Hinton 2002) for

classification. Glorot et al. (2011) adopted deep learning, which learns to extract an abstract meaningful representation for each review in an unsupervised fashion. They used stacked denoising autoencoders as the building blocks of the deep network and trained a classifier based on the output of the network. Unlike other research, they only relied on the newly learnt features and did not adopt original word features. Our work also uses the latent space model for latent representation learning. The major differences are that we adopt the restricted Boltzmann machine (RBM) for latent representation learning, and additionally, we perform sentiment classification over a hybrid representation combining both the latent representation and the sentiment features from preselected sentiment words.

There are also a number of works that explored domain adaptation under aspecific context, and it is worth mentioning here. Peddinti and Chintalapoodi (2011) performed sentiment analysis of Twitter by adaptation data from Blippr and IMDB movie reviews. They proposed two iterative algorithms based on expectation maximization and Rocchio support vector machine for filtering out noisy data. The experimental results showed that their approach was quite effective with an F-score of up to 0.9. Mejova and Srinivasan (2012) studied the problem of sentiment analysis across media streams. They created a data set consisting of data from blogs, reviews, and Twitter, and concluded that models trained on some social media sources are generalizable to others and that Twitter is the best source of training data. Since those works are restricted to a specific context, the approaches might not work in general cases.

2.3. Other Sentiment Analysis Tasks

Some other sentiment analysis tasks were also investigated in existing literature and are worth mentioning in the context of this particular research. For example, Ding et al. (2008), Hu and Liu (2004), and Liu et al. (2005) studied the problem of feature-based sentiment analysis, which first discovers the targets on which opinions have been expressed in a sentence, and then determines whether the opinions are positive, negative, or neutral (Liu 2010). Jindal and Liu (2006), Li et al. (2010), and Xu et al. (2011) examined the problem of comparative opinion mining. Jindal and Liu (2008) explored the problem of opinion spam. Lastly, Pang and Lee (2008) provided a comprehensive review of work in sentiment analysis.

3. Solution Overview

We are interested in determining text sentiment orientation when in-domain labeled data are unavailable. The major obstacle for simply borrowing labeled data from other domains is the word distribution discrepancies between domains. The domain that provides labeled data is often referred to as the source domain, whereas the target domain is the domain on which we would like to perform sentiment classification. However, this obstacle can be overcome if we could map text in the source domains and the target domain into a common space where those discrepancies vanish, or reduce, to a great extent. The latent space model, e.g., the RBM, could serve this purpose. The assumption we make is that the latent representations would be similar for texts with the same sentiment label, even though their word representations differ.

In addition to borrowing labeled data from other domains, unsupervised learning methods, where labeled data are unneeded, can be applied. The unsupervised method relies on preselected opinionated words and underperforms the in-domain supervised methods (Turney 2002). However, our intuition is that a combination of preselected opinionated words along with cross-domain latent representation would improve the accuracy of existing approaches.

Furthermore, the selection of source domain classification plays a significant role for cross-domain. However, it has been rarely mentioned in the literature. In this research, we propose two approaches: (1) the intelligent single source domain (ISSD) method, and (2) the multiple source domain (MSD) method. The former one refers to automatically selecting the most similar domain as the source domain and the latter one uses all domains.

At a high level, our method combines two sources of information: (a) sentiment information from other domains, referred to as source domains, and (b) sentiment information from a hand-picked opinionated word list. We first learn latent space representations for texts where inter-domain distribution variations disappear, or at least reduce to a great extent. The RBM is adopted for this purpose due to its recent prominent performance in text-related tasks (Larochelle and Bengio 2008). Unlabeled data from source domains and the target domain are required for representation learning but they are readily collectable. Next, we identify opinionated words and calculate the positive ratio and the negative ratio in each document taking advantage of a preselected opinionated word list. Finally, we combine the two features accounting for positive and negative proportions along with the newly learnt latent space representations and train classifiers over this hybrid feature space.

Our approach has several key characteristics that make it quite different from the existing cross-domain classification approaches: (a) We only use unigrams while all previous work selected both unigrams and bigrams, and we lemmatize the words before feeding

them to our system. Pang et al. (2002) suggest that unigram information turned out to be the most effective. The unigram features make our approach more efficient in terms of performance, whereas the lemmatization reduces the sparseness in the data. (b) We use sentiment information from a preselected opinionated word list in addition to labeled data from source domains and construct hybrid feature representations for classification while nearly all of the existing works on cross-domain sentiment classification rely on out-domain labeled data alone. (c) Unlike most of the existing work, we rely only on newly learnt features. (d) We adopt the restricted Boltzmann machine for latent representation learning and experimental results demonstrate its superiority.

4. Solution Details

In this section, we describe the architecture of our system, and the details of each component in the architecture. We will use the piece of text "*iPhone has good reception and excellent display*" as an example for illustrative purposes throughout the rest of the paper.

4.1. System Architecture

The overall architecture of our approach is depicted in Figure 1. In the preprocessing step, we perform routine text processing procedures, including lemmatization and unigrams extraction. The domain selection refers to choosing the appropriate domain as the source domain. Feature construction aims to build the features for classification. It contains three components: (1) the latent features learning aims to learn latent representation; (2) the opinionated features expansion is responsible for building sentiment word features; and (3) the hybrid features construction combines these two sets of features. Lastly, we detect sentiment orientation using supervised machine learning methods. We describe each of these components in detail next.

4.2. Preprocessing

This section introduces the text processing procedure before inputting the data into the system.

4.2.1. Lemmatization. Before feeding the text data into our system, we first carry out lemmatization on each document using the Stanford core natural language processing (NLP) toolkit[1] on both labeled data from multiple source domains and test data from the target domain. Lemmatization, which transfers inflected forms to base form, or lemma, reduces the sparseness of the data and has been shown to be effective in text classification (Joachims 1998). For example, "runs," "ran," and "running" will all be converted to "run." Lemmatization is closely related to stemming. The difference is that stemming operates on a single word *without* knowledge of the context. For instance, the word "meeting" can either be a base form of a noun or an inflected form of a verb. Lemmatization will determine this based on the contextual part-of-speech (POS) information, and thus, it is more appropriate for our classification context.

4.2.2. Unigrams Extraction. In this work, we select only unigrams as training features, whereas all previous research considered both unigrams and bigrams. Experimental results of Pang et al. (2002) suggest that unigram information turned out to be the most effective, and none of the alternative features, e.g., bigrams, provides consistently better performance. With fewer features, our system can run more efficiently, especially for latent representation learning which is computationally expensive. We consider only the presence/absence of a word; the frequency of the word is not under consideration. The former achieves better results as shown in Pang et al. (2002). Furthermore, stop words, such as "a," "do," "be," are excluded since they are not helpful for our classification task.

Following the example under consideration, we will have "iPhone," "good," "reception," "excellent," and "display" after this preprocessing step.

4.3. Source Domain Selection

Selection of source domains plays an important role in domain adaptation. In this paper, we propose two approaches: (1) the ISSD method, and (2) the MSD method. The former one refers to automatically selecting the most similar domain as the source domain while the latter one uses data from all domains. So this step is only for the ISSD method, since data from all domains will be used for the MSD method. We will discuss which approach of using the source domain is better in the evaluation section.

As we discussed before, the reduction of the accuracy is because of the discrepancy between the source domain and the target domain. So we believe that the classification would be higher if the discrepancy is less. Kullback–Leibler divergence (KLD) (Kullback and Leibler 1951) is widely used to calculate the divergence between two probability distributions. It can be calculated as follows:

$$D_{\mathrm{KL}}(S\|T) = \sum_{w_i} S(w_i) \times \log \frac{S(w_i)}{T(w_i)}, \tag{1}$$

where $S(w_i)$ is the probability of word w_i appearing in the source domain and $T(w_i)$ is the probability of word w_i appearing in the target domain. However, KL divergence is asymmetric and undefined if $T(w_i) = 0$. To overcome these limitations,

[1] http://nlp.stanford.edu/downloads/corenlp.shtml (last accessed March 14, 2013).

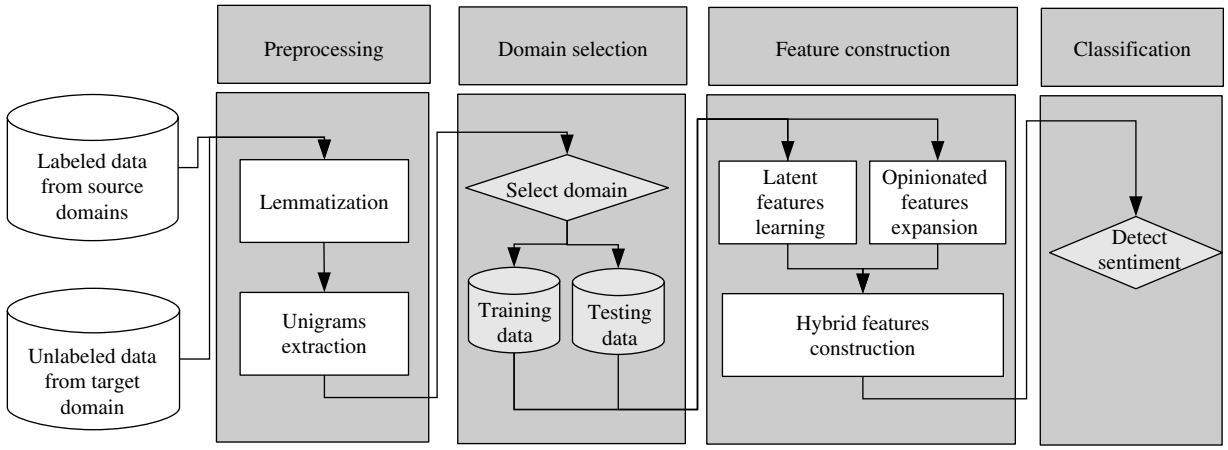

Figure 1 System Architecture

we adopt the Jensen–Shannon divergence (JSD) (Lin 1991) to measure the similarity between the source domain and the target domain. It is symmetric and measures the KLD between S, T, and the average of those two distributions:

$$\mathrm{JSD}(S\|T) = \tfrac{1}{2}D_{\mathrm{KL}}(S\|M) + \tfrac{1}{2}D_{\mathrm{KL}}(T\|M), \qquad (2)$$

where $M = \frac{1}{2}(S+T)$. The domain that has the lowest JSD with the target domain will be selected as the source domain.

4.4. Feature Construction

In this section, we elaborate on the procedure of feature construction.

4.4.1. Latent Features Learning. Any joint probability model that uses vectors of latent variables to abstract away from hand-crafted features whose format is designed by human, e.g., bigrams, would work for our latent representation learning step. The assumption is that the texts with the same sentiment label would have similar abstract representations where cross-domain distribution variation disappears, or at least will be reduced to a great extent, even though their text representations differ. Through the training, different words with the same sentiment from different domains, like "compact" (electronic domain) and "realistic" (video game domain), would correspond to the same latent variable. Therefore, the sentimental information is "transferred" from the source domain to the target domain. By using the newly learned representation, the feature representation discrepancy between source and target domain is reduced, which improves the classification performance.

In this research, we choose to use the RBM to learn latent and more abstract representations due to its recent prominent performance in text-related tasks (Larochelle and Bengio 2008). The RBM is an energy-based graphic model that associates a scalar energy to each configuration of the variables of interest and learning the parameters corresponds to modifying the energy function so that it has desired properties; e.g., we would like to have desirable configurations to have low energy. The RBM consists of a layer of hidden units and a layer of visible units. An RBM with three hidden units and four visible units is shown in Figure 2.

Suppose that an RBM models a distribution between n hidden units $\mathbf{h} = (h_1, h_2, \ldots, h_n)$ and d-dimension input visible units $\mathbf{v} = (v_1, v_2, \ldots, v_d)$. The energy function of the RBM is defined as

$$E(\mathbf{v}, \mathbf{h}) = -\mathbf{h}^T\mathbf{W}\mathbf{v} - \mathbf{b}^T\mathbf{v} - \mathbf{c}^T\mathbf{h}, \qquad (3)$$

where $\mathbf{W}$ represents the weights connecting hidden and visible units, and $\mathbf{b}$ and $\mathbf{c}$ are the offsets of the visible and hidden units, respectively.

Because of the specific structure of the RBM, visible and hidden units are conditionally independent given one another. In addition, both hidden units $\mathbf{h}$ and visible units $\mathbf{v}$ are binary in our context. So we can write the transition probability between the visible layer and the hidden layer as follows:

$$P(\mathbf{h} = 1 \mid \mathbf{v}) = \mathrm{sigm}(\mathbf{c} + \mathbf{W}\mathbf{v}), \qquad (4)$$

$$P(\mathbf{v} = 1 \mid \mathbf{h}) = \mathrm{sigm}(\mathbf{b} + \mathbf{W}^T\mathbf{h}), \qquad (5)$$

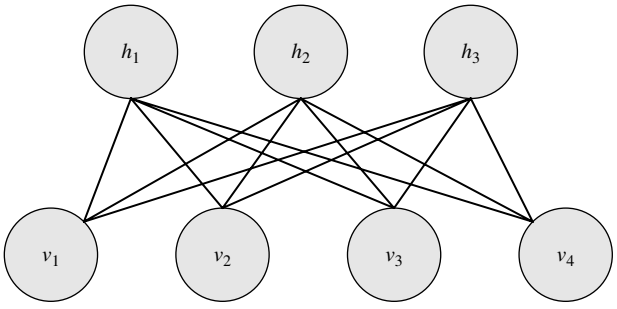

Figure 2 An RBM with Three Hidden Units and Four Visible Units

where sigm is the sigmoid function defined as

$$f(a) = \frac{1}{1+e^{-a}}. \tag{6}$$

The probability of a specific configuration is

$$P(\mathbf{v}, \mathbf{h}) \propto e^{-E(\mathbf{v}, \mathbf{h})}, \tag{7}$$

which allows us to write

$$P(\mathbf{v}) = \sum_{\mathbf{h}} e^{-E(\mathbf{v}, \mathbf{h})}. \tag{8}$$

The RBM can be trained by minimizing the empirical negative log-likelihood of the training data, and the cost function is

$$c(\mathbf{v}) = -\log P(\mathbf{v}). \tag{9}$$

Stochastic gradient descent is properly applied in the training process. However, in this research, we use contrastive divergence that can train the RBM much more efficiently (Carreira-Perpinan and Hinton 2005). The RBM is trained in an unsupervised manner, thus only unlabeled data are needed and they are readily collectable. Unlabeled data from both multiple source domains and the target domain are required. They are processed according to the procedures in the previous section before feeding into RBM training.

After learning the parameters, we convert the text representation of a document into a latent representation. Each visible variable represents a word with binary values; that is, "1" stands for presence and "0" otherwise. Using the learnt parameters and Equation (2), we can calculate the probabilities of each hidden variable being 1. Here we have two ways of constructing latent features. First, we can sample a value for each hidden variable given its probability and then take all hidden unit values as the feature vector to represent a specific document. Second, we can directly use the values of probabilities as latent representation. Either way will produce the same classification accuracy. In this paper, we choose the second way. For instance, if we choose the size of latent representation to be 5, the previous example would be converted into the likes of ("0.24," "0.79," "0.41," "0.94," "0.31").

4.4.2. Opinionated Features Expansion. Sentiment orientation can also be identified in an unsupervised manner. One simple example would be identifying orientation based on the ratio of the number of positive versus the number of negative words. If the ratio exceeds 1, the document might be positive; and vice versa.

We would like to combine this opinionated words feature with our latent space representation. Two features accounting for opinionated words in a document are (i) the ratio of the number of positive words versus the number of the total opinionated words, and (ii) the ratio of the number of negative words versus the total number of opinionated words. We use a list of positive and negative opinion words for calculation of these two ratios. The list is compiled over many years starting from 2004 by the author of Liu (2010) and consists of approximately 6,800 words.[2] Use of two ratios may seem a little duplicated since either value can be inferred by the other one. However, two-feature representation is necessary. Suppose we have only the positive ratio feature. The following two occasions would have the same value 0: (a) no opinionated word exists; (b) all the opinionated words are negative. Clearly these two cases are different and need to be distinguished. However, if we use two features, this will not be a problem. The first case is represented as $(0, 0)$ and the latter one is $(0, 1)$. In addition, if the number of positive words equals that of negative words, this representation will be 0.5 for both positive and negative features.

There are, of course, more sophisticated uses of opinionated words in literature. We only use the simplest one here and it is enough for performance improvement as will be shown in the experiment. Our example has two positive words ("good" and "excellent") and no negative words, so the opinionated words feature is (1 and 0), where the former value is the positive ratio and the latter corresponds to the negative ratio.

4.4.3. Hybrid Features Construction. To take advantage of both representations, we combine the two sets of features, i.e., latent features and opinionated words features, and form the hybrid feature representations. Following our example, after this step, we will have ("0.24," "0.79," "0.41," "0.94," "0.31," "1," "0") as the final representation.

4.5. Classification

At this stage we have hybrid representations for both training and testing data. The standard supervised machine learning methods can be applied easily. In this paper, we select the support vector machine (SVM) (Press et al. 2007) for sentiment classification; however, other classifiers can also be applied here. We first use the multiple source domain labeled data to train the SVM model on the basis of this hybrid representation, after which we use the hybrid representation of the target domain to classify the target documents as positive or negative sentiment.

5. Evaluation

In this section, we first describe our data set and evaluation metrics, and then discuss our experimental results.

[2] http://www.cs.uic.edu/~liub/FBS/opinion-lexicon-English.rar (last accessed March 14, 2013).

Table 1 Data Statistics

	Number of reviews		
Domain	Positive	Negative	Unlabeled
Books	1,000	1,000	4,465
DVDs	1,000	1,000	3,586
Electronics	1,000	1,000	5,681
Kitchen	1,000	1,000	5,945

5.1. Experimental Setting

The multidomain sentiment data set[3] is used in all existing work, and we will also use this data set for ease of comparison. The data set is collected by the authors of Blitzer et al. (2007). The multidomain sentiment data set contains product reviews taken from Amazon.com for many product types (domains). Each domain has 1,000 positive reviews, 1,000 negative reviews, and a number of unlabeled reviews—some domains (books and DVDs) have hundreds of thousands while others (musical instruments) have only a few hundred. Each review consists of a rating (0–5 stars), the reviewer's name, the reviewer's location, the product name, the review title, date, and the review text. Reviews with rating >3 were labeled positive, those with rating <3 were labeled negative, and the rest discarded because their polarity was ambiguous. In addition, a number of unlabeled reviews are also available for each domain.

All existing cross-domain sentiment classification research selected four domains: books, DVDs, electronics, and kitchen appliances. For ease of comparison, we will also evaluate our method over these four domains. The data statistics are listed in Table 1.

Similar to the previous research (Blitzer et al. 2007), we randomly select 200 positive reviews and 200 negative reviews as test data for each domain, and the remaining 1,600 labeled reviews in each domain are used as training data. All unlabeled data are used for latent representation learning. For computational reasons, only the top 5,000 frequent unigrams in the data set are selected as features for latent space representation learning.

The restricted Boltzmann machine was implemented using MATLAB.[4] In latent space model learning, we tried an extensive set of learning parameters and the following combination gave us the best results: hidden units: 5,000, learning rate 0.1, epochs: 30. The complete list of parameters is in Table 2. The SVM implemented in Weka (Hall et al. 2009) was selected as our classifier. When training SVMs, we chose the radial basis function kernel (Buhmann 2003) since we found that it consistently outperformed other counterparts in our classification context.

[3] http://www.cs.jhu.edu/~mdredze/datasets/sentiment/ (last accessed March 14, 2013).

[4] http://www.mathworks.com/products/matlab/ (last accessed March 14, 2013).

Table 2 Parameter Range

Parameter	Value
Learning rate	{0.1, 0.01, 0.001}
Epochs	{10, 15, 20, 25, 30}
Hidden units	{5,000}

5.2. Evaluation Metrics

We use two metrics to evaluate our method. The first one is accuracy, which captures the percentage of all reviews that are classified correctly. It can be computed as follows:

$$Accuracy = \frac{number\ of\ reviews\ correctly\ classified}{number\ of\ reviews\ in\ the\ test\ set}. \quad (10)$$

Accuracy is a widely used metric in literature and offers us direct Information on the performance of the classification. However, it incorporates the contribution of the classifier as well. To eliminate the effect of the classifier in the evaluation and assess the transfer efficiency more precisely, we adopt transfer loss that equals the reduction of accuracy compared with in-domain classification. This is quite necessary when we compare cross-domain sentiment classification methods using different classifiers. Let $e(S, T)$ be the error obtained by a method trained on the source domain S, or a combination of multiple source domains, and tested on the target domain T, and let $e(T, T)$ be the error of a method both trained and tested on target domain T using the same classifier, i.e., the in-domain method. Transfer loss can be calculated as follows:

$$L(S, T) = e(S, T) - e(T, T). \quad (11)$$

Transfer loss has been used in previous work (Blitzer et al. 2007, Glorot et al. 2011), and a lower value signifies a better performance.

5.3. Single Domain Method

This section presents the experimental results of cross-domain sentiment classification using a single source domain and validates our statement that use of similar domains as source domains would offer better results.

5.3.1. Domain Similarity. Each domain is represented by a 5,000-dimension vector and each dimension is valued by the probability of the corresponding word appearing in the domain. We calculated the probabilities based on the data set we used in the experiment. If a certain word does not appear in the domain, its probability is set to be 0. The Jensen–Shannon divergences between each pair of domains

Table 3 Domain Similarity

	Books	DVD	Electronics	Kitchen
Books		0.029730	0.361194	0.419062
DVD			0.196915	0.244196
Electronics				0.003937

Note. All values are in 10^{-4}.

Table 5 p-Values of Accuracy Significant Test for ISSD Method

Method	Unigram	RBM	In-domain
RBM	0.0079***		
Hybrid	0.0011***	0.0076***	0.8756^{+}
In-domain	0.0000***	0.3551^{+}	

$^{+}$Two-side test conducted.
***$p < 0.01$.

are then calculated and presented in Table 3. A lower value indicates less divergent, that is, more similar.

From Table 3, we can see that electronics and kitchen are quite similar since the divergence is quite small. According to the results, we would select DVD as the source domain for books and vice versa; and choose kitchen as the source domain for electronics and vice versa.

5.3.2. Accuracy. In the current research, there are three kinds of features: (1) unigrams, (2) latent (RBM), and (3) opinionated words ratios. The purpose of the current research is to propose a hybrid method for cross-domain sentiment classification that combines latent features and lexicon features. We can see the effectiveness of the latent features by comparing results of unigrams (1) and latent features (2) and show the effectiveness of the opinionated words features (3) by comparing results of latent features (2) and hybrid features (2 + 3). Thus, we only show the results for models trained on (1), (2), and (2) + (3) since we believe it is sufficient to achieve our research purpose.

Classification accuracies using only single source domains are presented in Table 4. The values of classification accuracy using the training data from the domain selected in the last section, i.e., the ISSD method, are in bold. From Table 4, we can see that, on average, the hybrid method outperforms the RBM method and the RBM method is superior to the unigrams method. These results demonstrate the effectiveness of our new representation.

The values in the first column of Table 4 are the results of the in-domain method where both training and testing data are from the same domain. The results of this method are considered as the gold standard for comparison. In the previous section, we select a source domain for each target domain and the corresponding results are better than their counterparts with only one exception (when in hybrid representation, using kitchen as the source domain provides better results than using books for the DVD domain). These results suggest that use of similar domains as the source domain could provide better results and Jensen–Shannon divergences can effectively measure the similarity.

We ran a series of t-tests to check if our latent space features are statistically more effective than those using word representations for the ISSD method. The p-values are as shown in Table 5.

From Table 5, we can see that the RBM method statistically outperformed the unigram method at the 0.01 level, and the hybrid method is significantly better than the RBM method and unigram method at the 0.01 level. In addition, the hybrid and RBM methods are not significantly different from the in-domain method, indicating that those two methods are as good as the in-domain method.

5.3.3. Transfer Loss. Transfer losses of the single source domain method are reported in Table 6. We follow the same structure as Table 6, and the transfer losses of the ISSD method are in bold. The average transfer losses for unigrams, RBM, and hybrid methods are 8.65, 5.73, and 3.62, respectively, which indicates that our representation learning could effectively reduce the reduction of accuracy. Transfer loss is less when a domain with less divergence is used as the source domain. One exception is using kitchen as the source domain which provides better results than using books for the DVD domain for hybrid representation. The results further confirm our statement that use of similar domains as source domains would provide better results.

Table 4 Classification Accuracy Using Single Source Domain

Source		Unigrams				RBM				Hybrid			
Target	In-domain	B	D	E	K	B	D	E	K	B	D	E	K
Books	83.00		**77.25**	69.00	70.00		**80.50**	73.00	73.50		**81.75**	76.75	75.25
DVD	81.50	**74.25**		70.50	73.00	**77.25**		75.25	77.25	**78.75**		79.50	81.00
Electronics	81.75	72.75	76.00		**78.00**	70.00	74.50		**83.25**	72.25	76.50		**83.75**
Kitchen	87.25	74.75	76.25	**85.00**		80.50	79.75	**87.00**		81.50	81.50	**88.50**	
Average	83.38		74.73				77.65				79.75		

Notes. All values are in percentages. ISSD results are in bold. B: books; D: DVD; E: electronics; K: kitchen.

Table 6 Transfer Loss Using Single Source Domain

Source	Unigrams				RBM				Hybrid			
Target	B	D	E	K	B	D	E	K	B	D	E	K
Books		**5.75**	14.00	13.00		**2.50**	10	9.5		**1.25**	6.25	7.75
DVD	**7.25**		11.00	8.5	**4.25**		6.25	4.25	**2.75**		2	0.5
Electronics	9.00	5.75		**3.75**	11.75	7.25		**−1.5**	9.5	5.25		**−2**
Kitchen	12.5	11.00	**2.25**		6.75	7.5	**0.25**		5.75	5.75	**−1.25**	
Average		8.65				5.73				3.62		

Notes. All values are in percentages. ISSD results are in bold. B: books; D: DVD; E: electronics; K: kitchen.

5.4. Multiple Domains Method

This section presents the experimental results of cross-domain sentiment classification using multiple source domains and compares the intelligent single source domain method and multiple source domains method.

5.4.1. Accuracy. Classification accuracies for various methods are presented in Table 7. Each row corresponds to results that one of the four domains serves as target domain. For instance, the first row presents results where books is the target domain. All values in the table are in percentages.

The first column of the table shows results of the method using opinionated words as the sole source. It classified the review as positive if the number of positive words surpass the number of negative words and negative otherwise. When these two numbers are equal, we set it as positive. The accuracies range from 70.35% to 76.85% with an average of 73.50%.

The middle part of the table corresponds to the ISSD method, that is, intelligently select the domain which is most similar with the target domain as the source domain. As we can see from the table, classification accuracy ranges from 74.25% to 85% with an average of 78.63% when unigrams are used as features. The average accuracy goes up to 82% when latent features are used and further increases to 83.75% when the two opinionated words features are included.

Results of the multiple source domain method are presented in the right part of the table. As we can see, the average accuracy of the multiple source domain method is higher than that of the intelligent single source domain method whatever feature is used. We postulate that the reason might be: (a) we can collect more data and a larger number of training instances would benefit classification; (b) word distributions in different domains vary and combination of multiple source domains will increase the probability that words in the test set behave less discordantly with respect to those in the training set.

Classification accuracy ranges from 75.25% to 82.75% with an average of 79.19% when unigrams are used as features. When using latent representations learnt by the RBM, the classification accuracy rises to 83.69% on average. In addition, it outperforms the in-domain method in the DVD domain and the electronics domain. This conclusively demonstrates the effectiveness of our latent representation learning. Finally, we train our classifiers over the hybrid representations, which combine the latent representations, and opinionated words features. The accuracy further steps up to 85.19% on average and ranges from 84.25% to 87.75%. It produces better results than the in-domain method in all four domains.

One interesting point is that the MSD is inferior to the ISSD when "kitchen" is treated as the target domain, no matter what set of features are used. We postulate the reason might be that the source domain of ISSD, "electronics," is quite similar with the kitchen domain (their JSD is quite close to 0 as reported in Table 3), and thus the ISSD provides good out-domain results. As we can see from Table 6, the transfer loss using unigrams is only 2.25, indicating that the out-domain result is close to the in-domain result. When we use the MSD, the "books" domain, which is quite different from the kitchen domain, is added for training and the accuracy is reduced. This result indicates that when we have a source domain which is quite similar with the target domain, it is better to use that domain as the sole source domain instead of use multiple source domain data simultaneously. From the series of results with kitchen as the target domain, we can see that our latent feature learning effectively reduces the discrepancy between source domain and target domain. From Table 7, we can see that when kitchen is the target domain, the MSD is 2.25% lower

Table 7 Classification Accuracy

Method		Intelligent single source domain (ISSD)			Multiple source domain (MSD)		
Target domain	Opinionated words	Unigrams	RBM	Hybrid	Unigrams	RBM	Hybrid
Books	70.35	77.25	80.50	81.75	75.25	82.00	84.25
DVD	73.75	74.25	77.25	78.75	77.75	83.50	84.50
Electronics	73.05	78.00	83.25	83.75	81.00	82.75	84.25
Kitchen	76.85	85.00	87.00	88.50	82.75	86.25	87.75
Average	73.50	78.63	82.00	83.75	79.19	83.69	85.19

Note. All values are in percentages.

Table 8 ***p*-Values of Accuracy Significant Test for MSD Method**

Method	Unigram	RBM	In-domain
RBM	0.0073***		
Hybrid	0.0080***	0.0045***	0.0250**
In-domain	0.0310**	0.7610+	

+Two-tail test result is reported.
**$p < 0.05$; ***$p < 0.01$.

than the ISSD for unigrams representation; whereas, when the RBM and hybrid representations are used, the MSD is only 0.75% lower than the ISSD. This further proves the effectiveness of our latent feature learning.

We also ran a series of *t*-tests to check if our latent space learning results are statistically better than those using word representations for the MSD method. For example, we want to know whether the superiority of the RBM method over the unigram method is statistically significant. We calculated the increase of accuracy for each domain and then ran a one-tail *t*-test to see whether the difference is statistically greater than 0. The *p*-values are reported in Table 8.

From the results in Table 8, we can conclude that the RBM and hybrid methods are statistically better than multiple sources with the unigram representation method at a 0.01 level. The hybrid method is statistically superior to the RBM method at a 0.01 level. The results in Table 3 suggest that our hybrid approach of combining the latent space learning and opinionated words features are effective.

In addition, the in-domain method is statistically better than the multiple sources method at the 0.05 level. The one-tail *t*-test results for the in-domain method over the RBM are not statistically significant, indicating that the in-domain method is not statistically better. So we report the two-tail *t*-test results to see if there is any difference between the in-domain method and the RBM method statistically, i.e., if the accuracy difference is equal to zero. Results in the table show that both differences are not statistically significant, indicating that the RBM method is statistically as good as the in-domain method. Furthermore, the results suggest that our hybrid method statistically outperforms the in-domain method at the 0.05 level.

Besides checking whether our new features are statistically effective, we also would like to know whether the superiority of the multiple sources domain method to the intelligent single source domain method is statistically significant. Again, we run a one-tail *t*-test and the *p*-value is 0.059, which indicates that the multiple sources domain method is statistically better than the intelligent single source domain method at the 0.1 level.

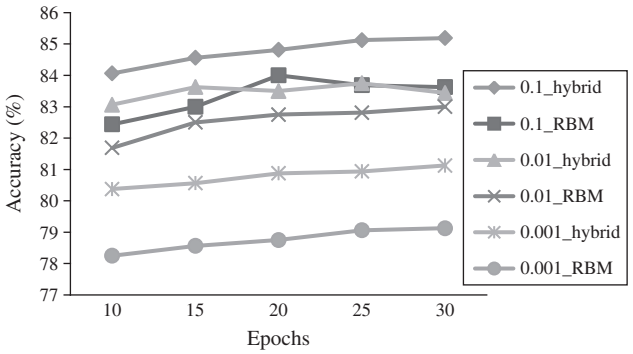

Figure 3 **Accuracy Curve**

We also report the average accuracies of the RBM and hybrid multiple source domains methods under different parameter settings shown in Figure 3. From the figure, we can see that the curves of the hybrid methods are always above the curves of the RBM methods with the same learning rate. In addition, the curves are upward sloping with few exceptions; that is, accuracies typically go up as the epochs increase. However, the slope is decreasing gradually. For example, in the curve of the hybrid method with learning rate 0.1, the line between epoch 25 and 30 is nearly flat. For the sake of space, we do not report graphs for the single source domain method.

5.4.2. Transfer Loss. Next, we report the transfer loss, which captures the reduction of accuracy due to the use of out-domain sources, to assess the transfer efficiency. The results are shown in Table 9. We follow the same structure as Table 7, where the first column presents results of the opinionated words method, the middle left part shows results of the single source domain method, and the right part illustrates accuracies of the multiple source domain approaches.

As we can see from Table 9, the transfer loss averaged 9.88 with range from 7.75 to 12.65 when opinionated words are used as the sole source.

The transfer losses reduce dramatically when we use the most similar domain as the source domain. Average transfer losses are 4.75, 1.38, and −0.38 for unigrams, RBM, and hybrid features, respectively.

Table 9 **Transfer Loss**

Method		Intelligent single source domain (ISSD)			Multiple source domain (MSD)		
Target domain	Opinionated words	Unigrams	RBM	Hybrid	Unigrams	RBM	Hybrid
Books	12.65	5.75	2.5	1.25	7.75	1.00	−1.25
DVD	7.75	7.25	4.25	2.75	3.75	−2.00	−3.00
Electronics	8.70	3.75	−1.5	−2	0.75	−1.00	−2.50
Kitchen	10.40	2.25	0.25	−1.25	4.50	1.00	−0.50
Average	9.88	4.75	1.38	0.19	4.19	−0.25	−1.81

Note. All values are in percentages.

When using latent features learnt by the RBM, the transfer loss for the electronics domain is below 0, which indicates that the accuracy is higher than that of the in-domain method. Furthermore, there are two domains with negative transfer when the hybrid representations are used: electronics and kitchen.

When multiple source domains are used, the average transfer losses further reduce. The transfer loss is 4.19 for unigrams representation. When we use latent representations learnt by the RBM, the average transfer loss drops significantly to -0.25 with values of two domains being below 0. Furthermore, the average transfer loss reduces to -1.81 when the hybrid representations are adopted and values of all four domains are lower than 0. A value of average transfer loss less than zero suggests that the overall performance is even better than the in-domain method.

We do not report the significant test for transfer loss as it would fall in the same range as in Table 7.

We also report the average transfer loss for different sets of parameters for the multiple source domains method in Figure 4. The figure suggests that the average transfer losses tend to decrease as the epochs increase with several exceptions. However, the improvement is relatively small, which is around 1% for each curve from epoch 10 to 30. Furthermore, under the same learning rate, the curve of the hybrid method always lies below that of the RBM method. For the sake of space, we do not report graphs for the single source domain method.

It is also interesting to compare our work with previous ones, where the same data set has been used. From the previously reported results, we calculate the average transfer loss for the following previous research: Blitzer et al. (2007), Pan et al. (2010), He et al. (2011), Bollegala et al. (2011), Titov (2011), and Glorot et al. (2011). The results of previous methods as well as our ISSD and MSD methods using hybrid features are shown in Figure 5. From the figure, we can see that both of our methods outperform all compared methods. The results conclusively demonstrate the superiority of our methods over all existing work.

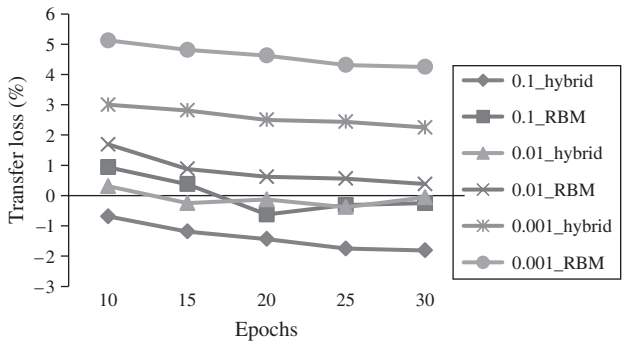

Figure 4 **Transfer Loss Curve**

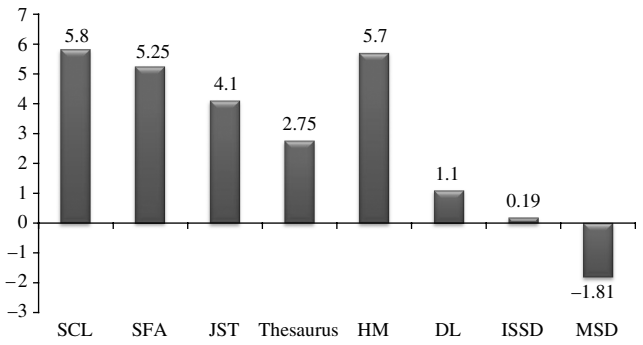

Figure 5 **Transfer Loss Across Methods**

6. Conclusion

In this paper, we proposed a novel framework for cross-domain sentiment classification using latent representation and opinionated words features. Specifically, our work has the following major contributions: (1) We utilized labeled data from the source domain and opinionated words. To the best of our knowledge, this research provides the first attempt to combine the sentiment information from source domain labeled data and hand-picked opinionated words together for the cross-domain sentiment classification task. (2) We propose to use Jensen–Shannon divergences to measure the domain similarity and use similar domains as the source domain. (3) Our experimental results show that our methods, both the ISSD and the MSD, statistically outperform the existing work addressing the same problem.

In the future, we plan to conduct a more thorough evaluation over a larger-scale data set with more domains. In addition, the simplest way of utilizing hand-picked opinionated words is used in our hybrid method. There are a number of much more sophisticated methods available in the literature. We are keen to see if an advanced method would further increase the accuracy.

References

Blitzer J, Dredze M, Pereira F (2007) Biographies, bollywood, boomboxes and blenders: Domain adaptation for sentiment classification. *Proc. 45th Annual Meeting Assoc. Comput. Linguistics* (*ACL'07*), *Prague, Czech Republic*, 187–205.

Bollegala D, Weir D, Carroll J (2011) Using multiple sources to construct a sentiment sensitive thesaurus for cross-domain sentiment classification. *Proc. 49th Annual Meeting Assoc. Comput. Linguistics* (*ACL'11*), *Portland, OR*, 132–141.

Buhmann MD (2003) *Radial Basis Functions*: *Theory and Implementations* (Cambridge University Press, Cambridge, UK).

Carreira-Perpinan MA, Hinton G (2005) On contrastive divergence learning. *Proc. Tenth Internat. Workshop on Artificial Intelligence Statist.* (*AISTATS'05*), *Savannah Hotel, Barbados*, 33–40.

Ding X, Liu B, Yu P (2008) A holistic lexicon-based approach to opinion mining. *Proc. 1st Conf. Web Search Web Data Mining* (*WSDM' 08*), *Palo Alto, CA*, 231–240.

Glorot X, Bordes A, Bengio Y (2011) Domain adaptation for large-scale sentiment classification: A deep learning approach. *Proc. 28th Internat. Conf. Machine Learn.* (*ICML'11*), *Bellevue, WA*, 513–520.

Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten I (2009) The WEKA data mining software: An update. *SIGKDD Explorations* 11(1):10–18.

Hatzivassiloglou V, Wiebe J (2000) Effects of adjective orientation and gradability on sentence subjectivity. *Proc. 18th Internat. Conf. Comput. Linguistics* (*COLING'00*), *Saarbrücken, Germany*, 174–181.

He Y, Lin C, Alani H (2011) Automatically extracting polarity-bearing topics for cross-domain sentiment classification. *Proc. 49th Annual Meeting Assoc. Comput. Linguistics* (*ACL'11*), *Portland, OR*, 123–131.

Hinton G (2002) Training products of experts by minimizing contrastive divergence. *Neural Comput.* 14(1):1771–1800.

Hu M, Liu B (2004) Mining and summarizing customer reviews. *Proc. 10th ACM Conf. Knowledge Discovery Data Mining* (*KDD'04*), *Seattle*, 168–177.

Jindal N, Liu B (2006) Identifying comparative sentences in text documents. *Proc. 29th Internat. ACM Conf. Res. Development Inform. Retrieval* (*SIGIR'06*), *Seattle*, 244–251.

Jindal N, Liu B (2008) Opinion spam and analysis. *Proc. 1st Conf. Web Search Web Data Mining* (*WSDM'08*), *Palo Alto, CA*, 219–230.

Joachims T (1998) Text categorization with support vector machines: Learning with many relevant features. *Proc. 10th Eur. Conf. Machine Learn.* (*ECML'98*), *Chemnitz, Germany*, 137–142.

Kullback S, Leibler R (1951) On information and sufficiency. *Ann. Math. Statist.* 22(1):79–86.

Larochelle H, Bengio Y (2008) Classification using discriminative restricted boltzmann machines. *Proc. 25th Internat. Conf. Machine Learn.* (*ICML'08*), *Helsinki, Finland*, 536–543.

Li S, Lin C-Y, Song Y-I, Li Z (2010) Comparable entity mining from comparative questions. *Proc. 48th Annual Meeting Assoc. Comput. Linguistics* (*ACL'10*), *Uppsala, Sweden*, 650–658.

Lin J (1991) Divergence measures based on the Shannon entropy. *IEEE Trans. Inform. Theory* 37(1):145–151.

Liu B (2010) Sentiment analysis and subjectivity. *Handbook of Natural Language Processing*, 2nd ed. (Chapman and Hall, Boca Raton, FL), 1–38.

Liu B, Hu M, Cheng J (2005) Opinion observer: Analyzing and comparing opinions on the Web. *Proc. 14th World Wide Web Conf.* (*WWW'05*), *Chiba, Japan*, 342–351.

Liu K, Zhao J (2009) Cross-domain sentiment classification using a two-stage method. *Proc. 18th ACM Conf. Inform. Knowledge Management* (*CIKM'09*), *Hong Kong*, 1717–1720.

Mejova Y, Srinivasan P (2012) Crossing media streams with sentiment: Domain adaptation in blogs, reviews and Twitter. *Proc. 6th Internat. AAAI Conf. Weblogs Soc. Media, Dublin, Ireland*, 234–241.

Pan SJ, Ni X, Sun J-T, Yang Q, Chen Z (2010) Cross-domain sentiment classification via spectral feature alignment. *Proc. 19th Internat. World Wide Web Conf.* (*WWW'10*), *Raleigh, NC*, 26–30.

Pang B, Lee L (2008) Opinion mining and sentiment analysis. *Foundations Trends Inform. Retrieval* 2(1):1–135.

Pang B, Lee L, Vaithyanathan S (2002) Thumbs up? Sentiment classification using machine learning techniques. *Proc. 2002 Conf. Empirical Methods Natural Language Processing* (*EMNLP'02*), *Philadelphia*, 79–86.

Peddinti V, Chintalapoodi P (2011) Domain adaptation in sentiment analysis of twitter. *Proc. 2011 AAAI Workshop Anal. Microtext, San Francisco*, 44–49.

Press W, Teukolsky S, Vetterling W, Flannery B (2007) Support vector machines. *Numerical Recipes*: *The Art of Scientific Computing*, 3rd ed. (Cambridge University Press, New York).

Riloff E, Wiebe J (2003) Learning extraction patterns for subjective expressions. *Proc. 2003 Conf. Empirical Methods Natural Language Processing* (*EMNLP'03*), *Sapporo, Japan*, 25–32.

Smolensky P (1986) Information processing in dynamical systems: Foundations of harmony theory. *Parallel Distributed Processing: Explorations in the Microstructures of Cognition* (MIT Press, Cambridge, MA), 194–281.

Titov I (2011) Domain adaptation by constraining inter-domain variability of latent feature representation. *Proc. 49th Annual Meeting Assoc. Comput. Linguistics* (*ACL'11*), *Portland, OR*, 417–424.

Turney P (2002) Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. *Proc. 40th Annual Meeting Assoc. Comput. Linguistics* (*ACL'02*), *Philadelphia*, 62–71.

Wiebet J, Bruce R (1999) Development and use of a gold standard data set for subjectivity classifications. *Proc. 27th Annual Meeting Assoc. Comput. Linguistics* (*ACL'99*), *College Park, MD*, 264–253.

Xu K, Liao SS, Li J, Song Y (2011) Mining comparative opinions from customer reviews for competitive intelligence. *Decision Support Systems* 51(4):743–754.