Leveraging Multiple Domains for Sentiment Classification

Fan Yang **Arjun Mukherjee** **Yifan Zhang**
Department of Computer Science, University of Houston, TX, USA
fyang11@uh.edu arjun4787@gmail.com aeryen@gmail.com

Abstract

Sentiment classification becomes more and more important with the rapid growth of user-generated content. However, sentiment classification task usually comes with two challenges: first, sentiment classification is highly domain-dependent and training sentiment classifier for every domain is inefficient and often impractical; second, since the quantity of labeled data is important for assessing the quality of classifier, it is hard to evaluate classifiers when labeled data is limited for certain domains. To address the challenges mentioned above, we focus on learning high-level features that are able to generalize across domains, so a global classifier can benefit with a simple combination of documents from multiple domains. In this paper, the proposed model incorporates both labeled and unlabeled data from multiple domains and learns new feature representations. Our model doesn't require labels from every domain, which means the learned feature representation can be generalized for sentiment domain adaptation. In addition, the learned feature representation can be used as classifier since our model defines the meaning of feature value and arranges high-level features in a prefixed order, so it is not necessary to train another classifier on top of the new features. Empirical evaluations demonstrate our model outperforms baselines and yields competitive results to other state-of-the-art works on the benchmark dataset.

1 Introduction

With the rapid growth of user-generated content, such as product reviews and microblogs, sentiment analysis and opinion mining have become more and more important as they address the problem of analyzing user's opinions, emotions, sentiments and attitudes. The applications of sentiment analysis have been found in almost every business and social domain (Liu, 2012; Bollen et al., 2011; Ku et al., 2006). Document-level sentiment classification predicts sentiment polarities for a given document or review. The large number of reviews not only help customers make better decisions but also make it possible yet challenge for product manufacturers to keep track opinions of the products (Hu and Liu, 2004).

While machine learning techniques provide interesting methods for analyzing sentiments (Turney, 2002; Go et al., 2009; Pang et al., 2002), challenges also arise certain limitations for the development of sentiment classification. For example, sentiment expression is highly domain-dependent (Pang and Lee, 2008), but training sentiment classifier for every domain is inefficient and often impractical. Simply combining data from different domains may not contribute to a generalized classifier for every domain, as users express the same sentiment in different domains using different words, or even express different sentiment using same words. For example, a user would prefer computer or car to "run fast" but not wish to use battery "die fast" or to see watch "move fast". A book or a movie can attract people by "unpredictable" endings but "unpredictable" economic trends scare away investors. In addition, certain domains don't have enough labeled data for building the classifier, which makes it indispensable to transfer knowledge between different domains.

Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 2978–2988, Osaka, Japan, December 11-17 2016.

A common observation is, even though sentiment expression is domain-dependent and various words are isolated by domain categories, there are always domain-independent words expressing general sentiment polarities. In traditional sentiment domain adaptation that focuses on one domain to another domain (Daumé III et al., 2010; Ben-David et al., 2007; Ben-David et al., 2010), such words are usually defined as pivot features (Blitzer et al., 2006; Blitzer et al., 2007; Pan et al., 2010). Existing works have focused on generating new feature representations for pivot features (Glorot et al., 2011; Chen et al., 2012; Yang and Eisenstein, 2015; Bollegala et al., 2015) so that classifiers trained on new features can generalize well across domains.

In this paper, we follow the motivation of using new feature representations (Bengio et al., 2012) to bridge domain divergence and transfer knowledge among domains. The idea of proposed work is to learn a high-level feature space where three constraints are enforced: the model can incorporate multiple domains with both labeled and unlabeled data; the high-level feature space maximizes the margin between sentiment polarities; the high-level feature can represent original features well so that two feature space can be transformed to each other through a shared parametric matrix. Some of the key characters of the proposed model are:

1. Given multiple domains, our model can leverage sentiment similarity between instances across different domains regardless of the dissimilarity between domains. This is achieved by maximizing the distance between sentiments and minimizing the distance between domains in the high-level feature space.

2. Compared with one domain(source) to another domain(target) schema, our model collaborates all possible domains with both labeled and unlabeled data, which is a more generic framework and caters for better transfer across domains.

3. Our model directly maximizes the margin of sentiment polarities in the learned feature space. This is achieved by exploiting non-linear transformation with sigmoid function and aligning instances to pseudo-sentiment centroids.

4. Unlike traditional representation learning method which involves two stages: learning representation and building classifier, the new feature space learned by our model can be taken as classifier by itself. This is achieved through setting the order of learned high-level features and defining the meaning of feature values. As a result, it is not necessary to train another classifier on top of the new features.

5. We extend autoencoder (Vincent et al., 2010) by incorporating sentiment polarities. Unlike existing semi-supervised autoencoder (Liu et al., 2015; Socher et al., 2011) that needs another layer for labels, our model introduces pseudo-sentiment centroids, which can be prefixed and selected without fine-tuning.

2 Related Work

In this section, we review related works on sentiment analysis and transfer learning.

2.1 In-Domain Sentiment Analysis

For sentiment analysis of user generated content, traditional works have focused on textual content and dictionaries based approaches (Taboada et al., 2011; Hu and Liu, 2004; Pang and Lee, 2008). Pang et al. (2002) built sentiment classifier to predict sentiment polarities of movie reviews. Hu et al. (2013) exploited contextual emotional signals for effective sentiment analysis in an unsupervised manner. Tumasjan et al. (2010) evaluated and analyzed Twitter messages with the political sentiment to predict the popularity of parties. Bollen et al. (2011) explored how Twitter mood patterns can identify economic events. Other trends of sentiment analysis are based on visual content (Siersdorfer et al., 2010; Borth et al., 2013) and multi-modalities (Socher et al., 2011). All these works consider training and testing data are within the same domain or following similar distributions.

Even though our work only focuses on textual content, we omit explicit word to word analysis but project word features into high-level feature space, where visual content or multi-modalities can also be transformed. In addition, we examine the limitation of domain-dependent sentiment expression and investigate efforts on building a generalized representation for all domains.

2.2 Knowledge Transfer and Leveraging Multiple Domains

Knowledge-based sentiment analysis have been explored in (Mukherjee and Liu, 2012; Liu, 2014; Chen et al., 2013a; Chen et al., 2013b; Chen et al., 2013c). However, they have focused on aspect term extraction as opposed to sentiment polarity extraction which is the focus of this work. Another thread is to transfer sentiment knowledge across domains. It is usually defined as domain adaptation (Daume III and Marcu, 2006; Ben-David et al., 2010; Ben-David et al., 2007). It utilizes the knowledge learned from one domain, referred as the source domain, to solve tasks in another domain, referred as the target domain. Studies have focused on re-weighting features that cross domains (Jiang and Zhai, 2007; Xia et al., 2013), using feature embeddings to convert word feature to vector feature (Yang and Eisenstein, 2015; Bollegala et al., 2015) or generating new feature representations that align domain-specified features onto a generalized feature space which can bridge domain divergence (Blitzer et al., 2006; Cheng and Pan, 2014). Blitzer et al. (2007) proposed Structural Correspondence Learning by selecting pivot feature and creating correlations between the pivot and non-pivot features. Pan et al. (2010) introduced a bipartite graph based approach to connect domain-independent features and domain-specific features. Xiao et al. (2013) proposed supervised word clustering, which assumed that a document was composed of latent (topical) clusters and used expectation-maximization algorithm to find those clusters to transform documents from bag-of-words representation to clusters representation.

Besides transferring knowledge from one domain to another domain, researchers have also explored the area of leveraging multiple domains (Mansour et al., 2009; Duan et al., 2009; Daumé III et al., 2010). In addition, Gong et al. (2012) introduced a kernel metric identifying the optimal adaptability among different domains. Li and Zong (2008) leveraged domains by combining sentiment classifiers of different domains to make the final prediction. Wu and Huang (2015) collaborated multiple domains by exploring textual content relations and sentiment word relations via labeled data. Glorot et al. (2011) utilized unlabeled data through unsupervised deep learning approach with rectifier. Chen et al. (2012) extended linear autoencoder by learning with marginalized corrupted features. Liu et al. (2015) incorporated domain and sentiment supervision for sentiment classification cross domains.

While the above works have made important progress, there are some major differences from this proposed work. First, instead of considering the transfer from "source" to "target", our model leverages multiple domains and at the time is capable of learning from both labeled and unlabeled data across multiple domains. This is closer to the reality because the amount of labels are various among domains and we want the model to leverage all possible knowledge. Second, our model exploits non-linear transformation with the sigmoid function. The sigmoid function shrinks feature value within $(0, 1)$, which enable us to directly maximize the margin of sentiment polarities by supervised instances alignment. Moreover, we fix the order of high-level features and define the meaning of feature values, so classification can be acquired by the learned representation. This is more efficient in terms of performance as it does not require retraining another classifier as other methods do.

3 The Proposed Model

The general idea of the proposed work is to learn a high-level feature space for multi-domain sentiment classification with three constraints. First, the collaboration constraint, which allows the model to collaborate multiple domains with both labeled and unlabeled data. Second, the max-min constraint, that employs high-level feature space to maximize the margin between sentiment polarities of instances across different domains and minimize the distance between domain clusters. Third, the transformation constraint where the high-level feature space and original feature space are transformed to each other through a shared weight matrix, which reduces overfitting.

3.1 Notations

Given a binary sentiment classification problem with positive and negative labels, we have access to P domains. There are total N documents, M of which are labeled. So for each domain j, documents and labels are denoted as $\{\mathbf{X}^j \in \mathbb{R}^{N_j \times D}, \mathbf{y}^j \in \mathbb{R}^{M_j \times 1}\}$, where D is the dimensions of the original feature space. We assume the original feature space is shared across different domains. $\mathbf{x}_i^j \in \mathbb{R}^{1 \times D}$ is the i^{th} document in domain j and represented as a Boolean vector of bag-of-words. If the i^{th} document is labeled, then $y_i^j \in \{+1, -1\}$. The shared weight matrix is denoted as $\mathbf{W}$. The bias vectors for transformation between original space and high-level feature space are denoted as $\mathbf{b}_1, \mathbf{b}_2$, separately.

3.2 The Model with Three Constraints

3.2.1 The Collaboration Constraint

The collaboration constraint enforces the model to collaborate multiple domains with all possible data. We adopt one layer denoising autoencoder (Vincent et al., 2010) to encourage this constraint and treat all data as unlabeled at this point. A denoising autoencoder learns high-level feature space $h(\mathbf{x})$. It corrupts input $\mathbf{x}$ and feeds the corrupted version $\bar{\mathbf{x}}$ into the encoding layer. The decoder undoes the corruption by generating results back to uncorrupted $\mathbf{x}$. The parameters of denoising autoencoder are learned by minimizing the reconstruction loss $\mathcal{L}_r\big(\mathbf{x}, g(\bar{\mathbf{x}})\big)$, where

$$g(\bar{\mathbf{x}}) = s_2(\mathbf{w}_2 h(\bar{\mathbf{x}}) + \mathbf{b}_2) \quad \text{and} \quad h(\bar{\mathbf{x}}) = s_1(\mathbf{w}_1 \bar{\mathbf{x}} + \mathbf{b}_1) \tag{1}$$

We utilize masking noise for the corruption and implement component wise logistic sigmoid as the non-linear function for both $s_1(\mathbf{x})$ and $s_2(\mathbf{x})$. It is important to have the value of each high-level feature of $h(\mathbf{x})$ between $\big(0, 1\big)$, as it paves the way for the following steps and makes the learned representation advisable for different classifiers. Note that the focus of this work is sentiment classification and we only utilize a single layer autoencoder, so the stack version of our implementation and the issue of handling the vanishing gradient are not discussed.

3.2.2 The Max-Min Constraint

The max-min constraint utilizes labeled data and supports the new representation $h(\mathbf{x})$ to maximize the margin of sentiment polarities. Sentiment classification is highly domain-dependent, which implies domain clusters are easier separated than sentiment clusters in the original feature space. While in $h(\mathbf{x})$, instances are aligned to prefixed sentiment pseudo-centroids if they have same sentiment polarities. Since the value of each high-level feature is between $\big(0, 1\big)$, we prefix $[1, \cdots, 1, 0, \cdots, 0]^T$ for positive pseudo-centroid $\mathbf{c}^+$ and $[0, \cdots, 0, 1, \cdots, 1]^T$ for negative pseudo-centroid $\mathbf{c}^-$, so the pseudo-centroids are maximized by cosine distance. This prefix also allows the learned representation to be used as classifier. The dimension of pseudo-centroid $|\mathbf{C}|$ is same as the dimension of learned feature space $|h(\mathbf{x})|$. After the alignment, the true sentiment centroids of labeled data would also be maximized. The alignment is achieved by minimizing the alignment loss $\mathcal{L}_a\big(\mathbf{C}, h(\mathbf{x})\big)$, where

$$\mathbf{C} = \begin{cases} \mathbf{c}^+, \text{if the label of } \mathbf{x} \text{ is positive} \\ \mathbf{c}^-, \text{if the label of } \mathbf{x} \text{ is negative} \end{cases} \tag{2}$$

When maximizing sentiment polarities between instances, the distance between domains is also minimized as positive instances across domains are moving towards $\mathbf{c}^+$ and negative instances are moving towards $\mathbf{c}^-$. This alignment can suppress domain-specific features because domains would be hardly partitioned.

3.2.3 The Transformation Constraint

We share the weight matrix between encoding layer and decoding layer in equation 1, so the equations are updated to:

$$g(\bar{\mathbf{x}}) = sigmoid(\mathbf{W}^T h(\bar{\mathbf{x}}) + \mathbf{b}_2) \quad \text{and} \quad h(\bar{\mathbf{x}}) = sigmoid(\mathbf{W}\bar{\mathbf{x}} + \mathbf{b}_1) \tag{3}$$

By sharing the weight matrix, the transformation would be more robust and provide a better feature representation as it reduces overfitting. The shared weights can also be interpreted as a trade-off between suppressing and preserving domain-specific features. We cannot fully eliminate those features, as domain-specific features give the ability to reconstruct to the original feature space.

3.3 Loss Function and Optimization

We use sum of Bernoulli Cross Entropy for reconstruction loss $\mathcal{L}_r\big(\mathbf{x}, g(\bar{\mathbf{x}})\big)$ and alignment loss $\mathcal{L}_a\big(\mathbf{C}, h(\mathbf{x})\big)$. Combined with equation 2 and 3, the final loss function $\mathcal{L}$ is

$$\begin{aligned}\mathcal{L} =& \mathcal{L}_r\big(\mathbf{x}, g(\bar{\mathbf{x}})\big) + \alpha\mathcal{L}_s\big(\mathbf{C}, h(\mathbf{x})\big) \\ =& -\frac{1}{N}\sum_{i}^{N}\sum_{k}^{|\mathbf{x}|}\mathbf{x}_{i,k}\log g(\bar{\mathbf{x}}_i)_k + \big(1-\mathbf{x}_{i,k}\big)\log\big(1-g(\bar{\mathbf{x}}_i)_k\big) \\ & -\alpha\frac{1}{M}\sum_{j}^{M}\sum_{l}^{|\mathbf{C}|}\mathbf{C}_l\log h(\mathbf{x}_j)_l + \big(1-\mathbf{C}_l\big)\log\big(1-h(\mathbf{x}_j)_l\big)\end{aligned} \tag{4}$$

We use $\alpha = 1$ in the model[1]. The equation 4 is non-convex but can be optimized with gradient descend. The partial derivatives of $\mathcal{L}$ with respect to $\mathbf{b}_1, \mathbf{b}_2$ and $\mathbf{W}$ is

$$\frac{\partial\mathcal{L}}{\partial\mathbf{W}} = \frac{1}{N}\sum_{i}^{N}(h(\bar{\mathbf{x}}_i) + h(\bar{\mathbf{x}}_i)(1-h(\bar{\mathbf{x}}_i))^T\mathbf{W}\bar{\mathbf{x}}_i\big) \times \big(g(\bar{\mathbf{x}}_i) - \mathbf{x}_i\big)^T + \frac{\alpha}{M}\sum_{j}^{M}(h(\mathbf{x}_j) - \mathbf{C})\mathbf{x}_j^T \tag{5}$$

$$\frac{\partial\mathcal{L}}{\partial\mathbf{b}_1} = \frac{1}{N}\sum_{i}^{N}h(\bar{\mathbf{x}}_i)(1-h(\bar{\mathbf{x}}_i))^T\mathbf{W}\big(g(\bar{\mathbf{x}}_i) - \mathbf{x}_i\big) + \frac{\alpha}{M}\sum_{j}^{M}(h(\mathbf{x}_j) - \mathbf{C}) \tag{6}$$

$$\frac{\partial\mathcal{L}}{\partial\mathbf{b}_2} = \frac{1}{N}\sum_{i}^{N}\big(g(\bar{\mathbf{x}}_i) - \mathbf{x}_i\big) \tag{7}$$

The output of our model is the learned feature representation $h(\mathbf{x})$. This representation can be used as features for another classifier or can be used as a classifier by itself.

3.4 Using the Representation as Classifier

To use $h(\mathbf{x})$ as a classifier, we calculate the distance from data points to the prefixed pseudo-centroids $\mathbf{C}$. Data points closer to $\mathbf{c}^+$ should have first half of $h(\mathbf{x})$ closer to 1 and the second half closer to 0, while data points closer to $\mathbf{c}^-$ should have first half of $h(\mathbf{x})$ closer to 0 and second half closer to 1. Therefore, the predicted label of a data point is:

$$sgn\big(\sum_{i=0}^{|\mathbf{C}|/2-1}h(\mathbf{x})_i - \sum_{j=0}^{|\mathbf{C}|/2-1}h(\mathbf{x})_{j+\frac{|\mathbf{C}|}{2}}\big) \tag{8}$$

The classifier can also be interpreted as using the decision of multiple logistic regression models. For the first half of $h(\mathbf{x})$, we define values greater than 0.5 represent positive and smaller than 0.5 represent negative, while for the second half, we define values greater than 0.5 represent negative and smaller than 0.5 represent positive.

4 Experimental Evaluations

We first report results of multi-domain sentiment classification, comparing different methods with in-domain classifier and multi-domain classifier. Then, we extend our model to domain adaptation problem with multiple source domains and one target domain. Finally, we evaluate the model with different metrics for a better understanding. All SVM classifiers are implemented through linear LibSVM (Chang and Lin, 2011) without tuning other parameters.

[1]We have tested the impact of different values of α from 0.2 to 10 and found the value of α does not significantly affect the performance of our model. A larger α generally gives a slightly better result.

4.1 Dataset

We use the Amazon product reviews (Blitzer et al., 2007) as our experimental dataset. The dataset has been widely used in multi-domain sentiment classification and domain adaptation for sentiment classification. There are 22 domains and more than 300,000 reviews in this dataset. We conduct experiments on reviews of 4 domains: **B**ooks, **D**vd, **E**lectronics and **K**itchen. Each of selected domain has 1000 positive and 1000 negative labeled reviews and roughly 5000 unlabeled reviews. The top 5000 of frequent 1-gram and 2-gram features are selected, as low frequent features are usually related to domains. The experiments are conducted based on 5-folder cross-validation by randomly splitting the labeled data into 5 partitions with equal size and we report the average result[2].

4.2 Performance Evaluation

We report results of multi-domain sentiment classification. The compared methods are listed below:

SVM-ID, SVM-MD: SVM classifier used for in-domain(ID) sentiment classification, and multi-domain(MD) sentiment classification with document-level combination.

ClassifierFusion (Li and Zong, 2008): Multi-domain sentiment classification with classifier-level combination. For each domain, a classifier is trained and used for all domains. The final prediction is the combination of the predictions of each individual classifier.

T-SVM (Sindhwani and Keerthi, 2006): Transductive SVM with document-level combination. All unlabeled data are used during learning. This method explicitly shows the performance of introducing unlabeled data to an SVM classifier, so it can be interpreted as SVM-MD with both labeled and unlabeled data.

SDA (Glorot et al., 2011): Unsupervised denoising autoencoder for representation learning. The feature space for the final SVM classifier is the concatenation of the original feature space and the learned representation feature space.

SDA-DSS (Liu et al., 2015): Representation learning with domain and sentiment supervision. The original implementation only incorporated sentiment labels of one domain, so we extend the model to incorporate labeled data of 4 domains. Same as SDA, the feature space for the SVM classifier is the concatenation of the original feature space and the learned representation feature space.

Proposed-R: Using representation learned from the proposed model. The feature space of the final SVM classifier is only the learned representation feature space. The hyper-parameter are explored as follow and selected by cross-validation: a masking noise probability in $\{0,0.5,0.6,0.7,0.8\}$ for corrupted $\bar{\mathbf{x}}$; dimension of learned feature space $h(\mathbf{x})$ in $\{100,250,500\}$; L_2 regularization penalty on shared weight matrix $\mathbf{W}$ in $\{0, 10^{-4}, 10^{-3}, 10^{-1}, 1\}$; learning rate for gradient descent in $\{0.01, 0.03, 0.1, 0.3\}$. The implementation is through Theano (Bastien et al., 2012).

Proposed-C: The learned representation is used as classifier with Equation 8. Because we only focus on the learned representation without tuning the parameter of SVM, the result of Proposed-C is only presented for a reference, not to demonstrate it is better than other classifiers.

All models, except SVM-ID, utilize the training set of 4 domains together and then make prediction on the test data of each domain. All models, except SVM-ID, SVM-MD, and ClassifierFusion, are implemented through transductive inference to better leverage the unlabeled data. However, it could be easily extended to inductive inference as all models return the feature transformation matrix.

	SVM-ID (%)		SVM-MD (%)		ClassifierFusion (%)		T-SVM (%)		SDA (%)		SDA-DSS (%)		Proposed-R (%)		Proposed-C (%)	
	Ac	F1	Ac	F1	Ac	F1	Ac	F1	Ac	F1	Ac	F1	Ac	F1	Ac	F1
B	77.81	78.25	78.03	78.26	79.16	79.28	77.50	76.67	80.49	80.70	79.60	79.84	**81.57**	**81.84**	84.14	84.56
D	77.61	78.05	79.14	79.13	81.37	81.45	79.67	79.53	80.73	80.93	80.23	80.25	**83.41**	**83.69**	84.91	85.44
E	82.54	82.68	83.52	83.55	84.77	84.03	86.17	86.05	84.65	84.64	84.21	84.16	**87.01**	**87.00**	88.17	88.18
K	84.78	84.56	84.39	84.26	86.42	86.05	86.17	85.33	86.18	85.97	85.53	85.16	**88.07**	**88.07**	89.30	89.40
Ave.	80.68	80.89	81.27	81.33	82.96	82.70	82.38	81.90	83.01	83.06	82.39	82.35	**85.02**	**85.15**	86.63	86.90

Table 1: Performance on Accuracy and F1 for Multi-Domain Sentiment Classification Tasks

[2]Micro average F1 on positive class is reported in this paper

According to Table 1, we find that our method consistently outperforms all other competitors showing that leveraging multiple domains can provide better results for sentiment classification if dissimilarities between domains are taken care of. It also shows that arbitrarily combining data together with bag-of-word representation cannot guarantee better results compared to in-domain sentiment classification. Compared to ClassifierFusion, the results validate that unlabeled instances and transductive inference can improve sentiment classification. Compared to T-SVM, it can be concluded that learning a new representation would benefit a generalized sentiment classifier across domains. Compared to SDA and SDA-DSS, the improvements can be explained as credit of transforming with sigmoid function, maximizing margin of sentiment polarities in learned feature space and suppressing domain-specific features during representation learning.

4.3 Unsupervised Knowledge Transfer

When incorporating multi-domains with both labeled and unlabeled data, a possible scenario is certain domains have very limited or even zero labels. In the literature of domain adaptation, unsupervised knowledge transfer or unsupervised domain adaptation usually refers to the situation where certain domains have no labeled instance at all, so the unlabeled domain, usually denoted as target domain, has to borrow labels from other domains denoted as source domain (Daume III and Marcu, 2006). In this experiment, we check the ability of our model for unsupervised domain adaptation. We consider 3 source domains and 1 target domain, and remove labels of each target domain separately. Table 2 presents the results of training on 3 domains and testing on the other. More specifically, SVM-ID remains the same as in Table 1 and the result is interpreted as an upper-bound for unsupervised knowledge transfer, while all other models are adjusted to access the training set of 3 source domains and test on the test set of the target domain.

	SVM-ID (%)		SVM-MD (%)		ClassifierFusion (%)		T-SVM (%)		SDA (%)		SDA-DSS (%)		Proposed-R (%)		Proposed-C (%)	
	Ac	F1	Ac	F1	Ac	F1	Ac	F1	Ac	F1	Ac	F1	Ac	F1	Ac	F1
B	77.81	78.25	74.22	74.42	75.00	75.08	75.33	75.37	76.54	76.75	76.86	76.33	**78.07**	**78.15**	81.00	81.41
D	77.61	78.05	76.41	77.76	77.00	76.53	78.00	77.86	79.64	**79.86**	**78.78**	79.83	78.11	79.03	83.35	83.74
E	82.54	82.68	80.90	80.27	82.16	81.86	82.17	**82.48**	81.95	81.42	**82.40**	81.73	82.29	81.30	85.09	84.20
K	84.78	84.56	81.90	82.14	83.83	83.90	83.67	83.11	84.04	84.00	83.92	84.23	**84.37**	**84.79**	87.25	87.31
Ave.	80.68	80.89	78.36	78.65	79.50	79.34	79.79	79.71	80.54	80.26	80.49	80.53	**80.71**	**80.82**	84.17	84.16

Table 2: Performance on Accuracy and F1 for Unsupervised Domain Adaptation with Multiple Sources

Comparing with transfer from one source to one target (Glorot et al., 2011; Liu et al., 2015), we observer from Table 2 that arbitrarily combining data together decreases the best transfer performances of SDA and SDA-DSS, which suggests domain-specific features are hurting unsupervised transfer. Moreover, our model yields limited improvements this time. One reason could be SDA and SDA-DSS separate domain-dependent and domain-independent features and keep all features in the learned representation, while our model suppresses domain-dependent feature. However, in general, "domain-dependent" is a relative definition. A word "story" could be a domain-independent feature for Books and DVD but also could be a domain-dependent feature for Books and Kitchen. Therefore, suppressing domain-dependent features for multiple domains which works better in the previous task could be the reason that limits our model on this unsupervised domain adaptation task.

4.4 Performance on Additional Metrics

We assess our model with additional metrics for multi-domain sentiment classification: sensitivity of labeled instances, proxy-A-distance, performance on different classifiers and suppressed features.

Sensitivity of Labeled Instances In this experiment, we explore how the proportion of labels affects the model as the model collaborates both labeled and unlabeled data from multiple domains. We limit the accessible labels in $\{0, 100, 300, 500, 1000, 1500\}$ for each domain to learn the representation and repeat the experiment of multi-domain sentiment classification with Proposed-R. With 0 labels, the model would be similar to a fully unsupervised SDA (Glorot et al., 2011) implementation, except with sigmoid activation and a lower feature dimension. According to Figure 1, limiting labels decreases the

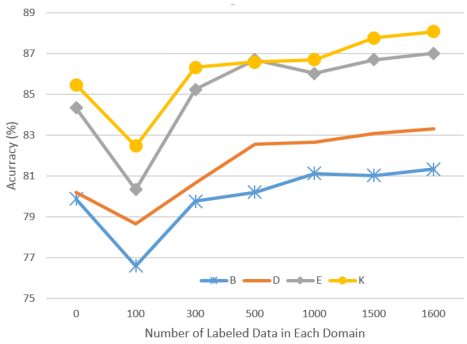

Figure 1: Sensitivity of Labeled Instances

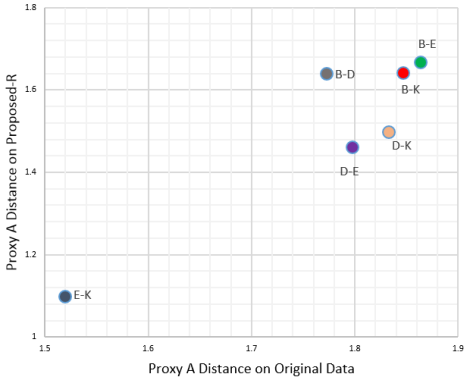

Figure 2: Proxy-A-Distance

performance, but this issue is generally solved as the proportion of labels is further increased and we see that at about 1000 labels performance starts to stabilize.

Proxy-A-Distance (PAD) We use PAD as an indirect metric to measure the ability to minimize dissimilarities between domains. The intuition is that removing domain-dependent features would weaken the discrimination among domains. The PAD metric (Ben-David et al., 2007) is defined as $2(1 - 2\epsilon)$, where ϵ is the generalization error and obtained by measuring how distinguishable are the two domains. In other words, we use the learned representation to accomplish domain recognition task. We randomly choose 1000 instances for training and 1000 for testing from each domain. Then we set up the recognition task as a binary classification problem with 6 combinations, for example, recognition between B and D. After applied our model, the PAD value of every recognition pair has decreased, which indicates the new feature representation learned from our model suppresses domain-dependent features.

According to Figure 2, recognizing Kitchen and Electronics are more difficult than Books and DVD. One reason is the reviews in Kitchen and Electronics are expressed using more domain-independent words. This observation also explains the sentiment classification results in Table 1 that classifying sentiment in Kitchen and Electronics generally have a better performance across different methods.

Performance of Different Classifiers We argue that a good representation should be able to benefit classification task without explicitly choosing or fine-tuning classifiers. Therefore, We repeat the experiment of multi-domain sentiment classification by comparing our Proposed-R(P-R) and SDA with another three state-of-art classifiers: K-Neighbors Classifier(KNC), Gaussian Naive Bayes(GNB) and RandomForest Classifier(RFC). According to Table 3, our model can still yield good results.

	KNC(%)		GNB(%)		RFC(%)	
	P-R	SDA	P-R	SDA	P-R	SDA
B	83.96	64.17	84.30	74.68	82.35	68.01
D	85.03	63.58	85.22	75.28	83.69	68.32
E	87.50	70.04	88.33	80.89	86.39	71.78
K	89.12	67.86	89.54	81.19	87.87	76.37

Table 3: Accuracy of Different Classifiers

Reconstructed	Original	Suppressed
not, poor, great, dont, was, after, bad, no, excellent, well, easy, best	was, my, so, num, you, all, one, if, very, great, good, just, not	since, way, all about, other, your time, them, when now, so, this_book

Table 4: Top Frequent Reconstruct Features, Original Features and Suppressed Features

Suppressed Features Our model has reconstructed 3785 features on average, compared to the original 5000, that means 1215 features are suppressed during the learning. We report some of the top frequent features in 3 ways: reconstructed by our model, in original space and suppressed by our model. From Table 4, the reconstructed feature are carrying more sentiment meaning than the original features, and the suppressed features involve domain-specific feature, such as "this_book", and non-sentiment features, such as "all", "so" and "your". This is what we expect by maximizing distance between sentiments in the learned representation feature space and sharing the transformation matrix between decoding and encoding layer.

5 Conclusion

This work proposed to leverage multiple domains with both labeled and unlabeled instances. The model learns high-level feature space with 3 constraints and achieves improved performance on multi-domain sentiment classification as attested by results on the benchmark dataset.

As our future work, we plan to explore multi-modality (e.g., mapping visual, video and sentiment content on the same space from multiple domains), and develop a recursive system where labeling work can be performed recursively with high confidence.

6 Acknowledgement

The authors would like to thank the anonymous reviewers for their comments. This work was supported in part by NSF 1527364.

References

Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, James Bergstra, Ian Goodfellow, Arnaud Bergeron, Nicolas Bouchard, David Warde-Farley, and Yoshua Bengio. 2012. Theano: new features and speed improvements. *arXiv preprint arXiv:1211.5590*.

Shai Ben-David, John Blitzer, Koby Crammer, Fernando Pereira, et al. 2007. Analysis of representations for domain adaptation. *Advances in neural information processing systems*, 19:137.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. 2010. A theory of learning from different domains. *Machine learning*, 79(1-2):151–175.

Yoshua Bengio, Aaron C Courville, and Pascal Vincent. 2012. Unsupervised feature learning and deep learning: A review and new perspectives. *CoRR, abs/1206.5538*, 1.

John Blitzer, Ryan McDonald, and Fernando Pereira. 2006. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 conference on empirical methods in natural language processing*, pages 120–128. Association for Computational Linguistics.

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

Danushka Bollegala, Takanori Maehara, and Ken-ichi Kawarabayashi. 2015. Unsupervised cross-domain word representation learning. *arXiv preprint arXiv:1505.07184*.

Johan Bollen, Huina Mao, and Alberto Pepe. 2011. Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. *ICWSM*, 11:450–453.

Damian Borth, Tao Chen, Rongrong Ji, and Shih-Fu Chang. 2013. Sentibank: large-scale ontology and classifiers for detecting sentiment and emotions in visual content. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 459–460. ACM.

Chih-Chung Chang and Chih-Jen Lin. 2011. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27.

Minmin Chen, Zhixiang Xu, Kilian Weinberger, and Fei Sha. 2012. Marginalized denoising autoencoders for domain adaptation. *arXiv preprint arXiv:1206.4683*.

Zhiyuan Chen, Arjun Mukherjee, Bing Liu, Meichun Hsu, Malu Castellanos, and Riddhiman Ghosh. 2013a. Discovering coherent topics using general knowledge. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 209–218. ACM.

Zhiyuan Chen, Arjun Mukherjee, Bing Liu, Meichun Hsu, Malu Castellanos, and Riddhiman Ghosh. 2013b. Exploiting domain knowledge in aspect extraction. In *EMNLP*, pages 1655–1667.

Zhiyuan Chen, Arjun Mukherjee, Bing Liu, Meichun Hsu, Malu Castellanos, and Riddhiman Ghosh. 2013c. Leveraging multi-domain prior knowledge in topic models. In *IJCAI*.

Li Cheng and Sinno Jialin Pan. 2014. Semi-supervised domain adaptation on manifolds. *IEEE transactions on neural networks and learning systems*, 25(12):2240–2249.

Hal Daume III and Daniel Marcu. 2006. Domain adaptation for statistical classifiers. *Journal of Artificial Intelligence Research*, 26:101–126.

Hal Daumé III, Abhishek Kumar, and Avishek Saha. 2010. Frustratingly easy semi-supervised domain adaptation. In *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*, pages 53–59. Association for Computational Linguistics.

Lixin Duan, Ivor W Tsang, Dong Xu, and Tat-Seng Chua. 2009. Domain adaptation from multiple sources via auxiliary classifiers. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 289–296. ACM.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 513–520.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1:12.

Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman. 2012. Geodesic flow kernel for unsupervised domain adaptation. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2066–2073. IEEE.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177. ACM.

Xia Hu, Jiliang Tang, Huiji Gao, and Huan Liu. 2013. Unsupervised sentiment analysis with emotional signals. In *Proceedings of the 22nd international conference on World Wide Web*, pages 607–618. ACM.

Jing Jiang and ChengXiang Zhai. 2007. Instance weighting for domain adaptation in nlp. In *ACL*, volume 7, pages 264–271.

Lun-Wei Ku, Yu-Ting Liang, Hsin-Hsi Chen, et al. 2006. Opinion extraction, summarization and tracking in news and blog corpora. In *AAAI spring symposium: Computational approaches to analyzing weblogs*, volume 100107.

Shoushan Li and Chengqing Zong. 2008. Multi-domain adaptation for sentiment classification: Using multiple classifier combining methods. In *Natural Language Processing and Knowledge Engineering, 2008. NLP-KE'08. International Conference on*, pages 1–8. IEEE.

Biao Liu, Minlie Huang, Jiashen Sun, and Xuan Zhu. 2015. Incorporating domain and sentiment supervision in representation learning for domain adaptation. In *Proceedings of the 24th International Conference on Artificial Intelligence*, pages 1277–1283. AAAI Press.

Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.

Zhiyuan Chen Arjun Mukherjee Bing Liu. 2014. Aspect extraction with automated prior knowledge learning.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. 2009. Domain adaptation with multiple sources. In *Advances in neural information processing systems*, pages 1041–1048.

Arjun Mukherjee and Bing Liu. 2012. Aspect extraction through semi-supervised modeling. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 339–348. Association for Computational Linguistics.

Sinno Jialin Pan, Xiaochuan Ni, Jian-Tao Sun, Qiang Yang, and Zheng Chen. 2010. Cross-domain sentiment classification via spectral feature alignment. In *Proceedings of the 19th international conference on World wide web*, pages 751–760. ACM.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 79–86. Association for Computational Linguistics.

Stefan Siersdorfer, Enrico Minack, Fan Deng, and Jonathon Hare. 2010. Analyzing and predicting sentiment of images on the social web. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 715–718. ACM.

Vikas Sindhwani and S Sathiya Keerthi. 2006. Large scale semi-supervised linear svms. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 477–484. ACM.

Richard Socher, Jeffrey Pennington, Eric H Huang, Andrew Y Ng, and Christopher D Manning. 2011. Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 151–161. Association for Computational Linguistics.

Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. 2011. Lexicon-based methods for sentiment analysis. *Computational linguistics*, 37(2):267–307.

Andranik Tumasjan, Timm Sprenger, Philipp Sandner, and Isabell Welpe. 2010. Predicting elections with twitter: What 140 characters reveal about political sentiment.

Peter D Turney. 2002. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 417–424. Association for Computational Linguistics.

Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. 2010. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, 11(Dec):3371–3408.

Fangzhao Wu and Yongfeng Huang. 2015. Collaborative multi-domain sentiment classification. In *Data Mining (ICDM), 2015 IEEE International Conference on*, pages 459–468. IEEE.

Rui Xia, Xuelei Hu, Jianfeng Lu, Jian Yang, Chengqing Zong, et al. 2013. Instance selection and instance weighting for cross-domain sentiment classification via pu learning. In *IJCAI*.

Min Xiao, Feipeng Zhao, and Yuhong Guo. 2013. Learning latent word representations for domain adaptation using supervised word clustering. In *EMNLP*, pages 152–162.

Yi Yang and Jacob Eisenstein. 2015. Unsupervised multi-domain adaptation with feature embeddings. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 672–682, Denver, Colorado, May–June. Association for Computational Linguistics.