

# Identifying Transferable Information Across Domains for Cross-domain Sentiment Classification

Raksha Sharma<sup>1</sup>, Pushpak Bhattacharyya<sup>1</sup>, Sandipan Dandapat<sup>2</sup> and Himanshu Bhatt<sup>2</sup>

<sup>1</sup>Department of Computer Science, Indian Institute of Technology Bombay

<sup>2</sup>Xerox Research Center of India, Bengaluru

<sup>1</sup>{raksha,pb}@cse.iitb.ac.in

<sup>2</sup>{sandipandandapat,bhathimanshu}@gmail.com

## Abstract

Getting manually labeled data in each domain is always an expensive and a time consuming task. Cross-domain sentiment analysis has emerged as a demanding concept where a labeled source domain facilitates a sentiment classifier for an unlabeled target domain. However, polarity orientation (positive or negative) and the significance of a word to express an opinion often differ from one domain to another domain. Owing to these differences, cross-domain sentiment classification is still a challenging task. In this paper, we propose that words that do not change their polarity and significance represent the transferable (usable) information across domains for cross-domain sentiment classification. We present a novel approach based on  $\chi^2$  test and cosine-similarity between context vector of words to identify polarity preserving significant words across domains. Furthermore, we show that a weighted ensemble of the classifiers enhances the cross-domain classification performance.

## 1 Introduction

The choice of the words to express an opinion depends on the domain as users often use domain-specific words (Qiu et al., 2009; Sharma and Bhattacharyya, 2015). For example, *entertaining* and *boring* are frequently used in the movie domain to express an opinion; however, finding these words in the electronics domain is rare. Moreover, there are words which are likely to be used across domains in the same proportion, but may change their polarity orientation from one domain to another (Choi et al., 2009). For example, a word like *unpredictable* is positive in the movie domain (*un-*

*predictable plot*), but negative in the automobile domain (*unpredictable steering*). Such a *polarity changing* word should be assigned positive orientation in the movie domain and negative orientation in the automobile domain.<sup>1</sup> Due to these differences across domains, a supervised algorithm trained on a labeled source domain, does not generalize well on an unlabeled target domain and the cross-domain performance degrades.

Generally, supervised learning algorithms have to be re-trained from scratch on every new domain using the manually annotated review corpus (Pang et al., 2002; Kanayama and Nasukawa, 2006; Pang and Lee, 2008; Esuli and Sebastiani, 2005; Breck et al., 2007; Li et al., 2009; Prabowo and Thelwall, 2009; Taboada et al., 2011; Cambria et al., 2013; Rosenthal et al., 2014). This is not practical as there are numerous domains and getting manually annotated data for every new domain is an expensive and time consuming task (Bhattacharyya, 2015). On the other hand, domain adaptation techniques work in contrast to traditional supervised techniques on the principle of transferring learned knowledge across domains (Blitzer et al., 2007; Pan et al., 2010; Bhatt et al., 2015). The existing transfer learning based domain adaptation algorithms for cross-domain classification have generally been proven useful in reducing the labeled data requirement, but they do not consider words like *unpredictable* that change polarity orientation across domains. Transfer (reuse) of changing polarity words affects the cross-domain performance negatively. Therefore, one cannot use transfer learning as the proverbial hammer, rather one needs to gauge what to transfer from the source domain to the target domain.

In this paper, we propose that the words which

<sup>1</sup>The word ‘unpredictable’ is a classic example of changing (inconsistent) polarity across domains (Turney, 2002; Fahrni and Klenner, 2008).

are equally significant with a consistent polarity across domains represent the usable information for cross-domain sentiment analysis.  $\chi^2$  is a popularly used and reliable statistical test to identify significance and polarity of a word in an annotated corpus (Oakes et al., 2001; Al-Harbi et al., 2008; Cheng and Zhulyn, 2012; Sharma and Bhattacharyya, 2013). However, for an unlabeled corpus no such statistical technique is applicable. Therefore, identification of words which are significant with a consistent polarity across domains is a non-trivial task. In this paper, we present a novel technique based on  $\chi^2$  test and cosine-similarity between context vector of words to identify Significant Consistent Polarity (SCP) words across domains.<sup>2</sup> The major contribution of this research is as follows.

1. *Extracting significant consistent polarity words across domains:* A technique which exploits cosine-similarity between context vector of words and  $\chi^2$  test is used to identify SCP words across labeled source and unlabeled target domains.
2. *An ensemble-based adaptation algorithm:* A classifier ( $C_s$ ) trained on SCP words in the labeled source domain acts as a seed to initiate a classifier ( $C_t$ ) on the target specific features. These classifiers are then combined in a weighted ensemble to further enhance the cross-domain classification performance.

Our results show that our approach gives a statistically significant improvement over Structured Correspondence Learning (SCL) (Bhatt et al., 2015) and common unigrams in identification of transferable words, which eventually facilitates a more accurate sentiment classifier in the target domain. The road-map for rest of the paper is as follows. Section 2 describes the related work. Section 3 describes the extraction of the SCP and the ensemble-based adaptation algorithm. Section 4 elaborates the dataset and the experimental protocol. Section 5 presents the results and section 6 reports the error analysis. Section 7 concludes the paper.

## 2 Related Work

The most significant efforts in the learning of transferable knowledge for cross-domain text clas-

sification are Structured Correspondence Learning (SCL) (Blitzer et al., 2007) and Structured Feature Alignment (SFA) (Pan et al., 2010). SCL aims to learn the co-occurrence between features from the two domains. It starts with learning pivot features that occur frequently in both the domains. It models correlation between pivots and all other features by training linear predictors to predict presence of pivot features in the unlabeled target domain data. SCL has shown significant improvement over a baseline (shift-unaware) model. SFA uses some domain-independent words as a bridge to construct a bipartite graph to model the co-occurrence relationship between domain-specific words and domain-independent words. Our approach also exploits the concept of co-occurrence (Pan et al., 2010), but we measure the co-occurrence in terms of similarity between context vector of words, unlike SCL and SFA, which literally look for the co-occurrence of words in the corpus. The use of context vector of words in place of words helps to overcome the data sparsity problem (Sharma et al., 2015).

Domain adaptation for sentiment classification has been explored by many researchers (Jiang and Zhai, 2007; Ji et al., 2011; Saha et al., 2011; Glorot et al., 2011; Xia et al., 2013; Zhou et al., 2014; Bhatt et al., 2015). Most of the works have focused on learning a shared low dimensional representation of features that can be generalized across different domains. However, none of the approaches explicitly analyses significance and polarity of words across domains. On the other hand, Glorot et al., (2011) proposed a deep learning approach which learns to extract a meaningful representation for each review in an unsupervised fashion. Zhou et al., (2014) also proposed a deep learning approach to learn a feature mapping between cross-domain heterogeneous features as well as a better feature representation for mapped data to reduce the bias issue caused by the cross-domain correspondences. Though deep learning based approaches perform reasonably good, they don't perform explicit identification and visualization of transferable features across domains unlike SFA and SCL, which output a set of words as transferable (reusable) features. Our approach explicitly determines the words which are equally significant with a consistent polarity across source and target domains. Our results show that the use of SCP words as features identified by our ap-

<sup>2</sup>SCP words are words which are significant in both the domains with consistent polarity orientation.

proach leads to a more accurate cross-domain sentiment classifier in the unlabeled target domain.

### 3 Approach: Cross-domain Sentiment Classification

The proposed approach identifies words which are equally significant for sentiment classification with a consistent polarity across source and target domains. These Significant Consistent Polarity (SCP) words make a set of transferable knowledge from the labeled source domain to the unlabeled target domain for cross-domain sentiment analysis. The algorithm further adapts to the unlabeled target domain by learning target domain specific features. The following sections elaborate SCP features extraction (3.1) and the ensemble-based cross-domain adaptation algorithm (3.2).

#### 3.1 Extracting SCP Features

The words which are not significant for classification in the labeled source domain, do not transfer useful knowledge to the target domain through a supervised classifier trained in the source domain. Moreover, words that are significant in both the domains, but have different polarity orientation transfer the wrong information to the target domain through a supervised classifier trained in the labeled source domain, which also downgrade the cross-domain performance.

Our algorithm identifies the significance and the polarity of all the words individually in their respective domains. Then the words which are significant in both the domains with the consistent polarity orientation are used to initiate the cross-domain adaptation algorithm. The following sections elaborate how the significance and the polarity of the words are obtained in the labeled source and the unlabeled target domains.

##### 3.1.1 Extracting Significant Words with the Polarity Orientation from the Labeled Source Domain

Since we have a polarity annotated dataset in the source domain, a statistical test like  $\chi^2$  test can be applied to find the significance of a word in the corpus for sentiment classification (Cheng and Zhulyan, 2012; Zheng et al., 2004). We have used *goodness of fit chi<sup>2</sup> test* with equal number of reviews in positive and negative corpora. This test is generally used to determine whether sample data

is consistent with a null hypothesis.<sup>3</sup> Here, the null hypothesis is that the word is equally used in the positive and the negative corpora. The  $\chi^2$  test is formulated as follows:

$$\chi^2(w) = ((c_p^w - \mu^w)^2 + (c_n^w - \mu^w)^2) / \mu^w \quad (1)$$

Where,  $c_p^w$  is the observed count of a word  $w$  in the positive documents and  $c_n^w$  is the observed count in the negative documents.  $\mu^w$  represents an average of the word's count in the positive and the negative documents. Here,  $\mu^w$  is the expected count or the value of the null-hypothesis. There is an inverse relation between  $\chi^2$  value and the  $p$ -value which is probability of the data given null hypothesis is true. In such a case where a word results in a  $p$ -value smaller than the critical  $p$ -value (0.05), we reject the null-hypothesis. Consequently, we assume that the word  $w$  belongs to a particular class (positive or negative) in the data, hence it is a significant word for classification (Sharma and Bhattacharyya, 2013).

**Polarity of Words in the Labeled Source Domain:** Chi-square test substantiates the statistically significant association of a word with a class label. Based on this association we assign a polarity orientation to a word in the domain. In other words, if a word is found significant by  $\chi^2$  test, then the exact class of the word is determined by comparing  $c_p^w$  and  $c_n^w$ . For instance, if  $c_p^w$  is higher than  $c_n^w$ , then the word is *positive*, else *negative*.

##### 3.1.2 Extracting Significant Words with the Polarity Orientation from the Unlabeled Target Domain

Target domain data is unlabeled and hence,  $\chi^2$  test cannot be used to find significance of the words. However, to obtain SCP words across domains, we take advantage of the fact that we have to identify significance of only those words in the target domain which are already proven to be significant in the source domain. We presume that a word which is significant in the source domain as per  $\chi^2$  test and occurs with a frequency greater than a certain threshold ( $\theta$ ) in the target domain is significant in the target domain also.

$$count_t(significant_s(w)) > \theta \Rightarrow significant_t(w) \quad (2)$$

Equation (2) formulates the significance test in the unlabeled target ( $t$ ) domain. Here, function

<sup>3</sup><http://stattrek.com/chi-square-test/goodness-of-fit.aspx?Tutorial=AP>.

$significant_s$  assures the significance of the word  $w$  in the labeled source ( $s$ ) domain and  $count_t$  gives the normalized count of the  $w$  in  $t$ .<sup>4</sup>  $\chi^2$  test has one key assumption that the expected value of an observed variable should not be less than 5 to be significant. Considering this assumption as a base, we fix the value of  $\theta$  as 10.<sup>5</sup>

**Polarity of Words in the Unlabeled Target Domain:** Generally, in a polar corpus, a positive word occurs more frequently in context of other positive words, while a negative word occurs in context of other negative words (Sharma et al., 2015).<sup>6</sup> Based on this hypothesis, we explore the contextual information of a word that is captured well by its context vector to assign polarity to words in the target domain (Rill et al., 2012; Rong, 2014). Mikolov et al., (2013) showed that similarity between context vector of words in vicinity such as ‘go’ and ‘to’ is higher compared to distant words or words that are not in the neighborhood of each other. Here, the observed concept is that if a word is positive, then its context vector learned from the polar review corpus will give higher cosine-similarity with a known positive polarity word in comparison to a known negative polarity word or vice versa. Therefore, based on the cosine-similarity scores we can assign the label of the known polarity word to the unknown polarity word. We term known polarity words as *Positive-pivot* and *Negative-pivot*.

**Context Vector Generation:** To compute context vector ( $conVec$ ) of a word ( $w$ ), we have used publicly available word2vec toolkit with the skip-gram model (Mikolov et al., 2013).<sup>7</sup> In this model, each word’s Huffman code is used as an input to a log-linear classifier with a continuous projection layer and words within a given window are predicted (Faruqui et al., 2014). We construct a 100 dimensional vector for each candidate word from the unlabeled target domain data. The decision method given in Equation

<sup>4</sup>Normalized count of  $w$  in  $t$  shows the proportion of occurrences of  $w$  in  $t$ .

<sup>5</sup>We tried with smaller values of  $\theta$  also, but they were not found as effective as  $\theta$  value of 10 for significant words identification.

<sup>6</sup>For example, ‘excellent’ will be used more often in positive reviews in comparison to negative reviews, hence, it would have more positive words in its context. Likewise, ‘terrible’ will be used more frequently in negative reviews in comparison to positive reviews, hence, it would have more negative words in its context.

<sup>7</sup> Available at: <https://radimrehurek.com/gensim/models/word2vec.html>

3 defines the polarity assignment to the unknown polarity words of the target domain. If a word  $w$  gives a higher cosine-similarity with the PosPivot (Positive-pivot) than the NegPivot (Negative-pivot), the decision method assigns the positive polarity to the word  $w$ , else negative polarity to the word  $w$ .

$$\begin{aligned} & \text{If}(\cosine(conVec(w), conVec(PosPivot)) > \\ & \quad \cosine(conVec(w), conVec(NegPivot))) \\ & \quad \Rightarrow \text{Positive} \\ & \text{If}(\cosine(conVec(w), conVec(PosPivot)) \\ & \quad < \cosine(conVec(w), conVec(NegPivot))) \\ & \quad \Rightarrow \text{Negative} \end{aligned} \quad (3)$$

**Pivot Selection Method:** We empirically observed that a polar word which has the highest frequency in the corpus gives more coverage to estimate the polarity orientation of other words while using context vector. Essentially, the frequent occurrence of the word in the corpus allows it to be in context of other words frequently. Therefore a polar word having the *highest frequency* in the target domain is observed to be more accurate as *pivot* for identification of polarity of input words.<sup>8</sup> Table 1 shows the examples of a few words in the electronics domain whose polarity orientation is derived based on the similarity scores obtained with PosPivot and NegPivot words in the electronics domain.

Word	Great	Poor	Polarity
Noisy	0.03	<b>0.24</b>	Neg
Crap	0.04	<b>0.28</b>	Neg
Weak	0.05	<b>0.21</b>	Neg
Defective	0.21	<b>0.70</b>	Neg
Sturdy	<b>0.43</b>	0.04	Pos
Durable	<b>0.44</b>	0.00	Pos
Perfect	<b>0.48</b>	0.20	Pos
Handy	<b>0.60</b>	0.21	Pos

Table 1: Cosine-similarity scores with PosPivot (great) and NegPivot (poor), and inferred polarity orientation of the words.

**Transferable Knowledge:** The proposed algo-

<sup>8</sup> To obtain the highest frequency based pivots, words in the target corpus (unlabeled) were ordered based on their frequency in the corpus, then a few top words were manually observed (by three human annotators) to pick out a positive word and a negative word. The positive and negative polarity of pivots were confirmed manually to get rid of random high frequency words (for example, *function words*). These highest frequency polar words were set as Positive-pivot and Negative-pivot.



Symbol	Description
$s, t$	Represent Source (s) and Target (t) respectively
$l, u$	Represent labeled and unlabeled respectively
$D_s^l, D_t^u$	Represent Dataset in $s$ and $t$ domains respectively
$V_s, V_t$	Vocabularies of words in the $s$ and $t$ respectively
$r_s^i, r_t^i$	$i^{th}$ review in $D_s^l$ and $D_t^u$ respectively
$sigPol()$	Identifies significant words with their polarity
$f$	Set of features
SVM	Implemented classification algorithm
$C_s$	Classifier $C_s$ is trained on $D_s^l$ with SCP as features
$R_t^n$	Top- $n$ reviews in $t$ as per classification score by $C_s$
$C_t$	Classifier $C_t$ is trained on $R_t^n$
$unigrams()$	Gives bag-of-words
$W_s, W_t$	Weights for $C_s$ and $C_t$ respectively
WSM	Weighted Sum Model

Table 2: Notations used in the paper

algorithm uses the above mentioned techniques to identify the significance and the polarity of words in the labeled source data (cf. Section 3.1.1) and the unlabeled target data (cf. Section 3.1.2). The words which are found significant in both the domains with the same polarity orientation form a set of SCP features for cross-domain sentiment classification. The weights learned for the SCP features in the labeled source domain by the classification algorithm can be reused for sentiment classification in the unlabeled target domain as SCP features have consistent impacts in both the domains.

### 3.2 Ensemble-based Cross-domain Adaptation Algorithm

Apart from the transferable SCP words (Obtained in Section 3.1), each domain has specific discriminating words which can be discovered only from that domain data. The proposed cross-domain adaptation approach (Algorithm 1) attempts to learn such domain specific features from the target domain using a classifier trained on SCP words in the source domain. An ensemble of the classifiers trained on the SCP features (transferred from the source) and domain specific features (learned within the target) further enhances the cross-domain performance. Table 2 lists the notations used in the algorithm. The working of the cross-domain adaptation algorithm is as follows:

1. Identify SCP features from the labeled source and the unlabeled target domain data.
2. A SVM based classifier is trained on SCP words as features using labeled source domain data, named as  $C_s$ .
3. The classifier  $C_s$  is used to predict the labels for the unlabeled target domain instances  $D_t^u$ ,

and the confidently predicted instances of  $D_t^u$  form a set of pseudo labeled instances  $R_t^n$ .

4. A SVM based classifier is trained on the pseudo labeled target domain instances  $R_t^n$ , using unigrams in  $R_t^n$  as features to include the target specific words, this classifier is named as  $C_t$ .
5. Finally, a Weighted Sum Model (WSM) of  $C_s$  and  $C_t$  gives a classifier in the target domain.

The confidence in the prediction of  $D_t^u$  is measured in terms of the classification-score of the document, *i.e.*, the distance of the input document from the separating hyper-plane given by the SVM classifier (Hsu et al., 2003). The top  $n$  confidently predicted pseudo labeled instances ( $R_t^n$ ) are used to train classifier  $C_t$ , where  $n$  depends on a threshold that is empirically set to  $|\pm 0.2|$ .<sup>9</sup> The classifier  $C_s$  trained on the SCP features (transferred knowledge) from the source domain and the classifier  $C_t$  trained on self-discovered target specific features from the pseudo labeled target domain instances bring in complementary information from the two domains. Therefore, combining  $C_s$  and  $C_t$  in a weighted ensemble (WSM) further enhances the cross-domain performance. Algorithm 1 gives the pseudo code of the proposed adaptation approach.

---

**Input:**  $D_s^l = \{r_s^1, r_s^2, r_s^3, \dots, r_s^j\}$ ,  
 $D_t^u = \{r_t^1, r_t^2, r_t^3, \dots, r_t^k\}$ ,  
 $V_s = \{w_s^1, w_s^2, w_s^3, \dots, w_s^p\}$ ,  
 $V_t = \{w_t^1, w_t^2, w_t^3, \dots, w_t^q\}$

**Output:** Sentiment Classifier in the Target Domain

- 1:  $SCP = sigPol(D_s^l) \cap sigPol(D_t^u)$
- 2:  $C_s = \text{Train-SVM}(D_s^l)$ , where  $f = SCP$
- 3: Predict Label:  $C_s(D_t^u) \rightarrow D_t^l$
- 4: Select:  $R_t^n | \forall r_t^i \in D_t^u, C_s(r_t^i) > \phi$ , where  $i \in \{1, 2, \dots, k\}$  and  $n \leq k$
- 5:  $C_t = \text{Train-SVM}(R_t^n)$ , where  $f = \{unigrams(R_t^n)\}$
- 6:  $WSM = (C_s * W_s + C_t * W_t) / (W_s + W_t)$
- 7: Sentiment Classifier in the Target Domain = WSM

**ALGORITHM 1:** Building of target domain classifier from the source domain

**Weighted Sum Model (WSM):** The weighted ensemble of classifiers helps to overcome the er-

<sup>9</sup>Balamurali et al., (2013) have shown that 350 to 400 labeled documents are required to get a high accuracy classifier in a domain using supervised classification techniques, but beyond 400 labeled documents there is not much improvement in the classification accuracy. Hence, threshold on classification score is set such that it can give a sufficient number of documents for supervised classification. Threshold  $|\pm 0.2|$  gives documents between 350 to 400.

rors produced by the individual classifier. The formulation of WSM is given in step-6 of the Algorithm 1. If  $C_s$  has wrongly predicted a document at boundary point and  $C_t$  has predicted the same document confidently, then weighted sum of  $C_s$  and  $C_t$  predicts the document correctly or vice versa. For example, a document is classified by  $C_s$  as negative (wrong prediction) with a classification-score of  $-0.07$ , while the same document is classified by  $C_t$  as positive (correct prediction) with a classification-score of  $0.33$ , the WSM of  $C_s$  and  $C_t$  will classify the document as *positive* with a classification-score of  $0.12$  (Equation 4).

$$WSM = \frac{(-0.07 * 0.765 + 0.33 * 0.712)}{(0.765 + 0.712)} = 0.12 \quad (4)$$

Here  $0.765$  and  $0.712$  are the weights  $W_s$  and  $W_t$  to the classifiers  $C_s$  and  $C_t$  respectively.

**Weights to the Classifiers in WSM:** The weights  $W_s$  and  $W_t$  are the classification accuracies obtained by  $C_s$  and  $C_t$  respectively on the cross-validation data from the target domain. The weights  $W_s$  and  $W_t$  allow  $C_s$  and  $C_t$  to participate in the WSM in proportion of their accuracy on the cross-validation data. This restriction facilitates the domination of the classifier which is more accurate.

#### 4 Dataset & Experimental Protocol

In this paper, we show comparison between SCP-based domain adaptation (our approach) and SCL-based domain adaptation approach proposed by Bhatt et al. (2015) using four domains, *viz.*, Electronics (E), Kitchen (K), Books (B), and DVD.<sup>10</sup> We use SVM algorithm with linear kernel (Tong and Koller, 2002) to train a classifier in all the mentioned classification systems in the paper. To implement SVM algorithm, we have used the publicly available Python based Scikit-learn package (Pedregosa et al., 2011).<sup>11</sup> Data in each domain is divided into three parts, *viz.*, train (60%), validation (20%) and test (20%). The SCP words are extracted from the training data. The weights  $W_s$  and  $W_t$  for the source and target classifiers are essentially accuracies obtained by  $C_s$  and  $C_t$

<sup>10</sup>The same multi-domain dataset is used by Bhatt et al. (2015), available at: <http://www.cs.jhu.edu/~mdredze/datasets/sentiment/index2.html>.

<sup>11</sup>Available at: <http://scikit-learn.org/stable/modules/svm.html>.

respectively on validation dataset from the *target domain*. We report the accuracy for all the systems on the test data. Table 3 shows the statistics of the dataset.

Domain	No. of Reviews	Avg. Length
Electronic (E)	2000	110 words
Kitchen (K)	2000	93 words
Books (B)	2000	173 words
DVD (D)	2000	197 words

Table 3: Dataset statistics

#### 5 Results

In this paper, we compare our approach with Structured Correspondence Learning (SCL) and common unigrams. SCL is used by Bhatt et al., (2015) for identification of transferable information from the labeled source domain to the unlabeled target domain for cross-domain sentiment analysis. They showed that transferable features extracted by SCL provide a better cross-domain sentiment analysis system than the transferable features extracted by Structured Feature Alignment (Pan et al., 2010). The SCL-based sentiment classifier in the target domain proposed by Bhatt et. al., (2015) is state-of-the-art for cross-domain sentiment analysis. On the other hand, common unigrams of the source and target are the most visible transferable information.<sup>12</sup>

**Gold standard SCP words:** Chi-square test gives us significance and polarity of the word in the corpus by taking into account the polarity labels of the reviews. Application of chi-square test in both the domains, considering that the target domain is also labeled, gives us gold standard SCP words. There is no manual annotation involved.

**F-score for SCP Words Identification Task:** The set of SCP words represent the usable information across domains for cross-domain classification, hence we compare the F-score for the SCP words identification task obtained with our approach, SCL and common-unigrams in Figure 1. It demonstrates that our approach gives a huge improvement in the F-score over SCL and common unigrams for all the 12 pairs of the source and target domains. To measure the statistical significance of this improvement, we applied *t*-test on the F-score distribution obtained with our approach, SCL and common unigrams. *t*-test is a

<sup>12</sup>Common unigrams is a set of unique words which appear in both the domains.

statistical significance test. It is used to determine whether two sets of data are significantly different or not.<sup>13</sup> Our approach performs significantly better than SCL and common unigrams, while SCL performs better than common unigrams as per  $t$ -test.

**Comparison among  $C_s$ ,  $C_t$  and WSM:** Table 4 shows the comparison among classifiers obtained in the target domain using SCP given by our approach, SCL, common-unigrams, and gold standard SCP for electronics as the source and movie as the target domains. Since electronics and movie are two very dissimilar domains in terms of domain specific words, unlike, books and movie, getting a high accuracy classifier in the movie domain from the electronics domain is a challenging task (Pang et al., 2002). Therefore, in Table 4 results are reported with electronics as the source domain and movie as the target domain.<sup>14</sup> In all four cases, there is difference in the transferred information from the source to the target, but the ensemble-based classification algorithm (Section 3.2) is the same. Table 4 depicts sentiment classification accuracy obtained with  $C_s$ ,  $C_t$  and WSM. The weights  $W_s$  and  $W_t$  in WSM are normalized accuracies by  $C_s$  and  $C_t$  respectively on the validation set from the target domain. The fourth column (size) represents the feature set size. We observed that WSM gives the highest accuracy, which validates our assumption that a weighted sum of two classifiers is better than the performance of individual classifiers. The WSM accuracy obtained with SCP words given by our approach is comparable to the accuracy obtained with gold standard SCP words.

The motivation of this research is to learn shared representation cognizant of significant and polarity changing words across domains. Hence, we report cross-domain classification accuracy obtained with three different types of shared representations (transferable knowledge), viz., common-unigrams, SCL and our approach.<sup>15</sup> System-1, system-2 and system-3 in Table 5 show the final cross-domain sentiment classification accuracy obtained with WSM in the target domain

<sup>13</sup>The detail about the test is available at: [http://www.socialresearchmethods.net/kb/stat\\_t.php](http://www.socialresearchmethods.net/kb/stat_t.php).

<sup>14</sup>The movie review dataset is a balanced corpus of 2000 reviews. Available at: <http://www.cs.cornell.edu/people/pabo/movie-review-data/>

<sup>15</sup>The reported accuracy is the ratio of correctly predicted documents to that of the total number of documents in the test dataset.

	$W_s$ & $W_t$	Features	Size	Acc
$C_s$		SCP (Our approach)	296	75.0
$C_t$		Unigrams	4751	74.3
<b>WSM</b>	0.72 & 0.69	-	-	<b>77.5</b>
$C_s$		SCL	1000	66.8
$C_t$		Unigrams	4615	68.0
<b>WSM</b>	0.63 & 0.61	-	-	<b>69.3</b>
$C_s$		Common-Unigrams	2236	64.0
$C_t$		Unigrams	4236	64.0
<b>WSM</b>	0.62 & 0.58	-	-	<b>65</b>
$C_s$		SCP (Gold standard)	163	77.0
$C_t$		Unigrams	1183	78.5
<b>WSM</b>	0.73 & 0.75	-	-	<b>80.0</b>

Table 4: Classification accuracy in % given by  $C_s$ ,  $C_t$  and WSM with different feature sets for electronics as source and movie as target.

for 12 pairs of source and target using common-unigrams, SCL and our approach respectively.

**System-1:** This system considers common-unigrams of both the domains as shared representation. **System-2:** It differs from system-1 in the shared representation, which is learned using Structured Correspondence Learning (SCL) (Bhatt et al., 2015) to initiate the process. **System-3:** This system implements the proposed domain adaptation algorithm. Here, the shared representation is the *SCP words* and the ensemble-based domain adaptation algorithm (Section 3.2) gives the final classifier in the target domain. Table 5 depicts that the system-3 is better than system-1 and system-2 for all pairs, except K to B and B to D. For these two pairs, system-2 performs better than system-3, though the difference in accuracy is very low (below 1%).

To enhance the final accuracy in the target domain, Bhatt et al., (2015) performed iterations over the pseudo labeled target domain instances ( $R_t^n$ ). In each iteration, they obtained a new  $C_t$  trained on increased number of pseudo labeled documents. This process is repeated till all the training instances of the target domain are considered. The  $C_t$  obtained in the last iteration makes WSM with  $C_s$  which is trained on the transferable features given by SCL. Bhatt et al., (2015) have shown that iteration-based domain adaptation technique is more effective than one-shot

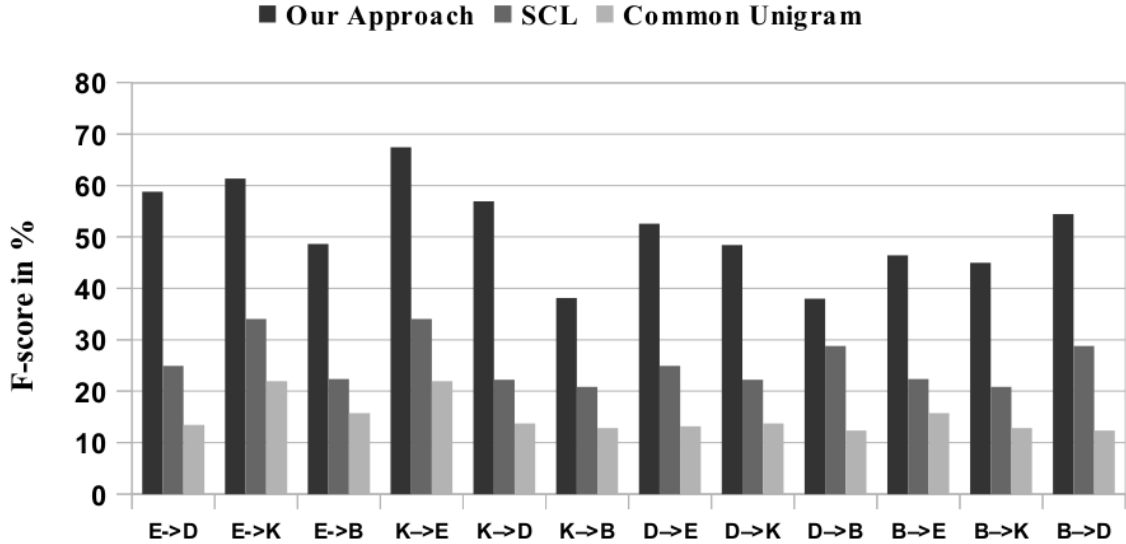


Figure 1: F-score for SCP words identification task (Source → Target) with respect to gold standard SCP words.

	System-1	System-2	System-3	System-4	System-5	System-6
D→B	62	64.2	<b>67</b>	66	76.5	<b>78.5</b>
E→B	63	58.9	<b>68.3</b>	67	75.6	<b>76.3</b>
K→B	67	<b>68.75</b>	67.85	69	71.2	<b>74</b>
B→D	76	<b>81</b>	80.5	77	81.5	<b>81.5</b>
E→D	68	71	<b>77.5</b>	71.5	74	<b>80.5</b>
K→D	69	69	<b>74</b>	71	75.2	<b>77</b>
B→E	68	66	<b>73</b>	69	79	<b>81.2</b>
D→E	61	62	<b>74</b>	66	73.2	<b>74.2</b>
K→E	76	75.75	<b>80</b>	78	81	<b>82</b>
B→K	66	67.5	<b>72</b>	69	79.2	<b>80.5</b>
D→K	65.75	67	<b>71</b>	66	80	<b>81</b>
E→K	74.25	75	<b>85.75</b>	76	84	<b>85.75</b>

Table 5: Cross-domain sentiment classification accuracy in the target domain (Source (S) → Target (T)).

Domain	Significant Words	Unigrams
Books (B)	76	89
DVD (D)	82.5	84
Electronics (E)	82.5	85
Kitchen (K)	82.5	86

Table 6: In-domain sentiment classification accuracy using significant words and unigrams.

adaptation approaches. **System-4**, **system-5**, and **system-6** in Table 5 incorporate the iterative process into **system-1**, **system-2**, and **system-3** respectively. We observed the same trend after the inclusion of the iterative process also, as the SCP-based system-6 performed the best in all 12 cases. On the other hand, SCL-based system-5 performs better than the common-unigrams based system-4. Table 7 shows the results of significance test (t-test) performed on the accuracy distributions produced by the six different systems. The notice-

able point is that the iterations over SCL (system-5) and our approach (system-6) narrow down the difference in the accuracy between system-2 and system-3 as system-2 and system-3 have a statistically significant difference in accuracy with the p-value of 0.039 (Row-4 of Table 7), but the difference between system-5 and system-6 is not statistically significant. Essentially, system-3 does not give much improvement with iterations, unlike system-2. In other words, addition of the iterative process with the shared representation given by SCL overcomes the errors introduced by SCL. On the other hand, SCP given by our approach were able to produce a less erroneous system in one-shot. Table 6 shows the in-domain sentiment classification accuracy obtained with unigrams and significant words as features considering labeled data in the domain. System-6 tries to equalize the in-domain accuracy obtained with unigrams.



	P-value	Significant?
sys1 vs sys2	0.719	No
sys1 vs sys3	0.011	Yes
sys2 vs sys3	0.039	Yes
sys1 vs sys4	0.219	No
sys2 vs sys4	0.467	No
sys3 vs sys4	0.090	No
sys1 vs sys5	0	Yes
sys2 vs sys5	0	Yes
sys3 vs sys5	0.101	No
sys4 vs sys5	0	Yes
sys1 vs sys6	0	Yes
sys2 vs sys6	0	Yes
sys3 vs sys6	0.0130	Yes
sys4 vs sys6	0	Yes
sys5 vs sys6	0.231	No

Table 7: t-test ( $\alpha = 0.05$ ) results on the difference in accuracy produced by various systems (cf. Table 5).

To validate our assertion that polarity preserving significant words (SCP) across source and target domains make a less erroneous set of transferable knowledge from the source domain to the target domain, we computed Pearson product-moment correlation between F-score obtained for our approach (cf. Figure 1) and cross-domain accuracy obtained with SCP (System-3, cf. Table 5). We observed a *strong positive correlation* ( $r$ ) of 0.78 between F-score and cross-domain accuracy. Essentially, an accurate set of SCP words positively stimulates an improved classifier in the unlabeled target domain.

## 6 Error Analysis

The pairs of domains which share a greater number of domain-specific words, result in a higher accuracy cross-domain classifier. For example, Electronics (E) and Kitchen (K) domains share many domain-specific words, hence pairing of such similar domains as the source and the target results into a higher accuracy classifier in the target domain. Table 5 shows that  $K \rightarrow E$  outperforms  $B \rightarrow E$  and  $D \rightarrow E$ , and  $E \rightarrow K$  outperforms  $B \rightarrow K$  and  $D \rightarrow K$ . On the other hand, DVD (D) and electronics are two very different domains unlike electronics and Kitchen, or DVD and books. The DVD dataset contains reviews about the music albums. This difference in types of reviews makes them to share less number of words. Table 8 shows the percent (%) of common words among the 4 domains. The percent of common unique words are common unique words divided by the summation

of unique words in the domains individually.

E - D	E - K	E - B	D - K	D - B	K - B
15	22	17	14	22	17

Table 8: Common unique words between the domains in percent (%).

## 7 Conclusion

In this paper, we proposed that the Significant Consistent Polarity (SCP) words represent the transferable information from the labeled source domain to the unlabeled target domain for cross-domain sentiment classification. We showed a strong *positive correlation* of 0.78 between the SCP words identified by our approach and the sentiment classification accuracy achieved in the unlabeled target domain. Essentially, a set of less erroneous transferable features leads to a more accurate classification system in the unlabeled target domain. We have presented a technique based on  $\chi^2$  test and cosine-similarity between context vector of words to identify SCP words. Results show that the SCP words given by our approach represent more accurate transferable information in comparison to the Structured Correspondence Learning (SCL) algorithm and common-unigrams. Furthermore, we show that an ensemble of the classifiers trained on the SCP features and target specific features overcomes the errors of the individual classifiers.

## References

- S Al-Harbi, A Almuhareb, A Al-Thubaity, MS Khorshed, and A Al-Rajeh. 2008. Automatic arabic text classification.
- AR Balamurali, Mitesh M Khapra, and Pushpak Bhattacharyya. 2013. Lost in translation: viability of machine translation for cross language sentiment analysis. In *Computational Linguistics and Intelligent Text Processing*, pages 38–49. Springer.
- Himanshu S. Bhatt, Deepali Semwal, and S. Roy. 2015. An iterative similarity based adaptation technique for cross-domain text classification. In *Proceedings of Conference on Natural Language Learning*, pages 52–61.
- Pushpak Bhattacharyya. 2015. *Multilingual Projections*. Springer International Publishing, Cham.
- John Blitzer, Mark Dredze, Fernando Pereira, et al. 2007. Biographies, bollywood, boom-boxes and

- blenders: Domain adaptation for sentiment classification. In *Proceedings of Association for Computational Linguistics*, pages 440–447.
- Eric Breck, Yejin Choi, and Claire Cardie. 2007. Identifying expressions of opinion in context. In *Proceedings of International Joint Conference on Artificial Intelligence*, pages 2683–2688.
- Erik Cambria, Bjorn Schuller, Yunqing Xia, and Catherine Havasi. 2013. New avenues in opinion mining and sentiment analysis. *IEEE Intelligent Systems*, (2):15–21.
- Alex Cheng and Oles Zhulyn. 2012. A system for multilingual sentiment learning on large data sets. In *Proceedings of International Conference on Computational Linguistics*, pages 577–592.
- Yoonjung Choi, Youngho Kim, and Sung-Hyon Myaeng. 2009. Domain-specific sentiment analysis using contextual feature generation. In *Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*, pages 37–44. ACM.
- Andrea Esuli and Fabrizio Sebastiani. 2005. Determining the semantic orientation of terms through gloss classification. In *Proceedings of International Conference on Information and Knowledge Management*, pages 617–624.
- Angela Fahrni and Manfred Klenner. 2008. Old wine or warm beer: Target-specific sentiment analysis of adjectives. In *Proc. of the Symposium on Affective Language in Human and Machine, AISB*, pages 60–63.
- Manaal Faruqui, Jesse Dodge, Sujay K Jauhar, Chris Dyer, Eduard Hovy, and Noah A Smith. 2014. Retrofitting word vectors to semantic lexicons. *arXiv preprint arXiv:1411.4166*.
- 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.
- Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin. 2003. A practical guide to support vector classification. Technical report, Department of Computer Science, National Taiwan University.
- Yang-Sheng Ji, Jia-Jun Chen, Gang Niu, Lin Shang, and Xin-Yu Dai. 2011. Transfer learning via multi-view principal component analysis. *Journal of Computer Science and Technology*, 26(1):81–98.
- Jing Jiang and ChengXiang Zhai. 2007. Instance weighting for domain adaptation in nlp. In *Proceedings of Association for Computational Linguistics*, pages 264–271.
- Hiroshi Kanayama and Tetsuya Nasukawa. 2006. Fully automatic lexicon expansion for domain-oriented sentiment analysis. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*, pages 355–363.
- Tao Li, Yi Zhang, and Vikas Sindhwani. 2009. A non-negative matrix tri-factorization approach to sentiment classification with lexical prior knowledge. In *Proceedings of International Joint Conference on Natural Language Processing*, pages 244–252.
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *CoRR*.
- Michael Oakes, Robert Gaaizauskas, Helene Fowkes, Anna Jonsson, Vincent Wan, and Micheline Beaulieu. 2001. A method based on the chi-square test for document classification. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 440–441. ACM.
- 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 International Conference on World Wide Web*, pages 751–760.
- 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 Conference on Empirical Methods in Natural Language Processing*, pages 79–86.
- Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):2825–2830.
- Rudy Prabowo and Mike Thelwall. 2009. Sentiment analysis: A combined approach. *Journal of Informetrics*, 3(2):143–157.
- Guang Qiu, Bing Liu, Jiajun Bu, and Chun Chen. 2009. Expanding domain sentiment lexicon through double propagation. In *IJCAI*, volume 9, pages 1199–1204.
- Sven Rill, Jörg Scheidt, Johannes Drescher, Oliver Schütz, Dirk Reinel, and Florian Wogenstein. 2012. A generic approach to generate opinion lists of phrases for opinion mining applications. In *Proceedings of the First International Workshop on Issues of Sentiment Discovery and Opinion Mining*, page 7. ACM.
- Xin Rong. 2014. word2vec parameter learning explained. *arXiv preprint arXiv:1411.2738*.

- Sara Rosenthal, Preslav Nakov, Alan Ritter, and Veselin Stoyanov. 2014. Semeval-2014 task 9: Sentiment analysis in twitter. In *Proceedings of SemEval*, pages 73–80.
- Avishek Saha, Piyush Rai, Hal Daumé III, Suresh Venkatasubramanian, and Scott L DuVall. 2011. Active supervised domain adaptation. In *Machine Learning and Knowledge Discovery in Databases*, pages 97–112.
- Raksha Sharma and Pushpak Bhattacharyya. 2013. Detecting domain dedicated polar words. In *Proceedings of the International Joint Conference on Natural Language Processing*, pages 661–666.
- Raksha Sharma and Pushpak Bhattacharyya. 2015. Domain sentiment matters: A two stage sentiment analyzer. In *Proceedings of the International Conference on Natural Language Processing*.
- Raksha Sharma, Mohit Gupta, Astha Agarwal, and Pushpak Bhattacharyya. 2015. Adjective intensity and sentiment analysis. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*.
- Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. 2011. Lexicon-based methods for sentiment analysis. *Computational Linguistics*, 37(2):267–307.
- Simon Tong and Daphne Koller. 2002. Support vector machine active learning with applications to text classification. *The Journal of Machine Learning Research*, 2:45–66.
- Peter D. Turney. 2002. Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of Association for Computational Linguistics*, pages 417–424.
- Rui Xia, Chengqing Zong, Xuelei Hu, and Erik Cambria. 2013. Feature ensemble plus sample selection: domain adaptation for sentiment classification. *IEEE Intelligent Systems*, 28(3):10–18.
- Zhaohui Zheng, Xiaoyun Wu, and Rohini Srihari. 2004. Feature selection for text categorization on imbalanced data. *ACM Sig KDD Explorations Newsletter*, 6(1):80–89.
- Joey Tianyi Zhou, Sinno Jialin Pan, Ivor W Tsang, and Yan Yan. 2014. Hybrid heterogeneous transfer learning through deep learning. In *AAAI*, pages 2213–2220.