Cross-Domain Sentiment Tagging Using Meta-Classifier and a High Accuracy In-Domain Classifier

Balamurali A R[1] **Debraj Manna**[2] **Pushpak Bhattacharyya**[2]
[1] IITB-Monash Research Academy, IIT Bombay
[2]Dept. of Computer and Science Engineering, IIT Bombay
Mumbai, India - 400076
{balamurali,debraj,pb}@cse.iitb.ac.in

Abstract

Drop in accuracy due to a shift in domain is common problem for all NLP tasks including sentiment tagging. In this paper, we propose an approach to improve cross domain sentiment tagging accuracy. The idea is to use a group of classifiers trained on the source domain to generate noisy tagged data for the target domain. A small amount of hand-labeled target domain data is then used to decide a confidence threshold for filtering out the noise. The remaining data which is tagged with a high confidence is then used to train a high accuracy sentiment tagger for the target domain. On a training domain similar to target domain, our system performs in par or even better than a classifier trained using in-domain data. Further, even in case of dissimilar domains, our system gives a high cross domain classification accuracy with an average improvement of 4.39% over the best baseline accuracy.

1 Introduction

A popular task under sentiment analysis which has been well studied deals with sentiment prediction at document level (Pang and Lee, 2008). **Sentiment prediction** can be defined as classification of documents based on its sentiment content. Both supervised and unsupervised approaches have been used to create sentiment classification models (Pang and Lee, 2008). Former approaches are preferred to the latter because of their high classification accuracy (Pang and Lee, 2008). However, such approaches require large amount of labeled data because of the domain specific nature of sentiment analysis (Aue and Gamon, 2005). We define the task of creating labeled data based on the polarity of the document, namely *positive* or *negative*, as **Sentiment tagging**. Manual creation such labeled data is an expensive and tedious process. Moreover, sensitivity of sentiment prediction models to time limits the utility of this laborious task to a specific time ((Read, 2005)).

A possible solution to this problem is to leverage the labeled data in an existing domain and use it to create labeled data for new domains which have no or meager training data. We define this process as **Cross Domain Sentiment Tagging (CDST)**. Having enough tagged data can solve most problems (if not all), related to sentiment prediction, as the classification accuracy in an in-domain setup is high (Pang and Lee, 2008). Sophisticated feature engineering can be used to attain almost 94% accuracy in an in-domain classification setup (Matsumoto et al., 2005). However, in a cross-domain setup, the accuracy of classifier deteriorates when source and target data are different (Aue and Gamon, 2005). The root cause of this problem is *unknown words*; Target domain may contain words which the classifiers have not seen during training from the source domain.

Our approach to tackle the problem of CDST follows the intuition that given a task of categorizing a document with respect to its sentiment content on an unknown domain, an individual tends to bear his *knowledge and experience* in a previously encountered similar domain. This cross domain knowledge is then combined with *common sense knowledge* that he is aware of, to make the final decision. This decision making skill of the individual then increases as he encounters more and more such instances from this new domain.

Proceedings of ICON 2010: 8th International Conference on Natural Language Processing
Macmillan Publishers, India. Also accessible from http://ltrc.iiit.ac.in/proceedings/ICON-2010

For example, consider the following book review - *"As usual, Robin Cook keeps you on the edge of your seat until the end. Excellent reading"*. An individual who is familiar with *Movie* reviews will be aware of sentiment related to *"edge of your seat"* and considering the general polarity of *"Excellent"*, he will judge the *Book* review as positive.

We use a combination of meta-classifier and a high accuracy in-domain classifier to translate the above intuition to tackle CDST. The meta-classifier classifier comprises a set of three classifiers which are trained on different collection of features from the same source domain.

1. The first classifier creates a model incorporating all the ***domain-pertinent information*** of the training domain using all words as features. The model would thus be able to capture the essence of *"edge of your seat"* according to the *domain it belongs*.

2. The second classifier creates a model using more generic features provided by a lexicon which we refer as ***Universal Sentiment Clues***. The lexicon consists of words with their most commonly used polarity.The *common sense knowledge* with respect to polarity of *"Excellent"* will thus be captured.

3. The third classifier is a rule based classifier, which takes into consideration all the words having ***prior polarity*** in a document. Prior polarity of a word is its polarity without considering the context. This is the most naive way of thinking when one does not have specialised domain knowledge. In the example, *"Excellent"* is a word with positive prior polarity.

A combined model encompassing all the above models is then created for the sentiment tagging of the target domain. As some of the instances tagged by this model may be noisy; only those instances whose probability of being correct is above a threshold, are selected. The threshold is determined with the help of a small amount of labeled target data. These selected instances are then used for creating a highly accurate in-domain classifier based on Information Gain Ratio based feature selection. The model thus learned is employed to label all the target data. We observe considerable improvement in the CDST accuracy using our approach irrespective of the lexical dissimilarity of domains.

The roadmap for the rest of paper is as follows: Section 2 explains related work. Section 3 and 4 describe our approach and system architecture respectively. Results and discussions are presented in section 5. Section 6 concludes the paper and points to future work.

2 Related Work

Cross Domain sentiment classification is a research area with lot of commercial significance. Here we outline few works which are related to this area. Aue and Gamon (2005) showed that cross domain classification accuracy is less compared to in-domain classification due to domain specific nature of sentiment analysis. They also suggested if more common features are present in training and target then sentiment prediction accuracy would be higher.

Blitzer *et.al* (2007) also pointed out that **similarity of domains** can assist in creating better classifiers. The main idea behind this algorithm (Structural Correspondence Learning, SCL) is finding correspondence of feature from different domains to the common features, pivot features, occurring in source and target domain. The original features and correspondences with pivot features are then together used to train the classifier. The target domain is represented in the same format and final prediction is done. A small number of labeled data from the target domain allows a model learned from one domain to adapt to a new domain. However, the number of labeled examples necessary for successful adaptation depends on the similarity between domains.

A group of classifiers based on different features are increasingly being used for this task and are seen to be promising(Aue and Gamon, 2005; Blitzer et al., 2007; Andreevskaia and Bergler, 2008).

Dasgupta and Ng (2009) proposed a semi-supervised approach for sentiment classification where they first mined the unambiguous reviews and then used them to classify the ambiguous reviews.

Apart from some common dimensions in the previous works, our work is unique based on the feature set and system architecture used. More than the individual components in the system, it is the procedure that adds uniqueness to our system. Table 1 summarizes the related work in the area of cross domain sentiment analysis and also

Dimension	Aue and Gamon (2005)	Blitzer et al. (2007)	Andreevskaia and Bergler (2008)	Our Approach
Labeled Target Data	Yes	Yes	Yes	Yes
Universal Sentiment Clues	No	No	No	Yes
Use of Prior Polarity	No	No	Yes	Yes
Group of classifiers	Yes	No	Yes	Yes

Table 1: Related Work and Comparison

contextualizes our work.

3 Our Approach

Our approach consists of three major steps:

1. A noisy tagged data of target domain is created using a group of classifiers trained on a source domain.

2. Highly probable correct instances, from this partly erroneous tagged set, are categorized as actual tagged data with help of few hand-labeled target domain data.

3. A high accuracy classifier is modeled after selecting appropriate features based on their information gain ratio and used to completely tag the target domain data.

4 System Architecture

The Figure 1 shows the system architecture. It comprises of two main components.

1. High Accuracy Classifier (HAC)

2. Intermediary Tagger System (ITS)

Subsequent subsection describes each modules and how they are combined to develop the complete system.

4.1 High Accuracy in-domain Classifier

In-domain sentiment prediction systems perform with high classification accuracy (Pang and Lee, 2008). We exploit this fact in our approach for CDST. We first define an approach for creating High Accuracy in-domain Classifier using simple low-level features. Linguistic and semantic features can be employed to create high accuracy classifiers but simple low level features ,which are highly discriminatory, can be very effective for creating superior classifiers.

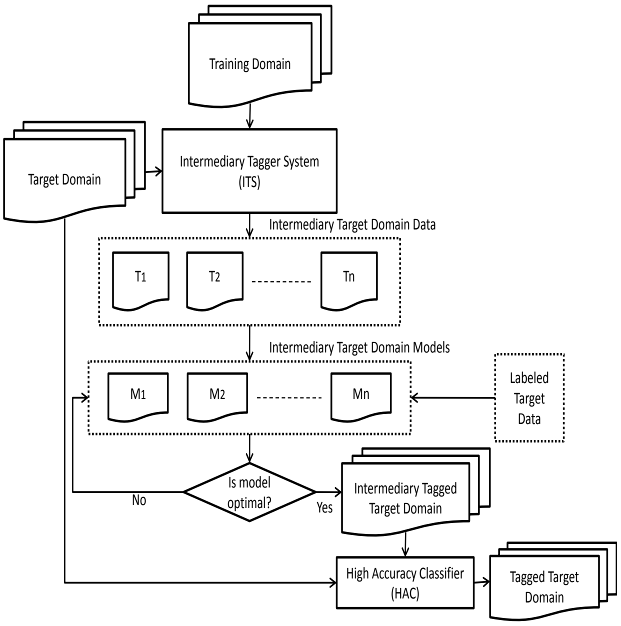
Training Domain
Target Domain
Intermediary Tagger System (ITS)
Intermediary Target Domain Data
T1
T2
Tn
Intermediary Target Domain Models
M1
M2
Mn
Labeled Target Data
No
Is model optimal?
Yes
Intermediary Tagged Target Domain
High Accuracy Classifier (HAC)
Tagged Target Domain

Figure 1: System Architecture

An **Information Gain Ratio based Feature Selection (IGR)** is done on the corpus to select domain pertinent features. We use the same principle as used by $ID3$, to select the best candidate features for our classification task (Mitchell, 1997). To avoid biasing towards attributes with many different values, a normalized form of information gain is used. Gain ratio normalizes the information gain over the entropy of that attribute.

The intuition for this approach is that people writing reviews on a particular product group tend to use the same and limited vocabulary to express their sentiment. As a result of this selection process, both sentiment and non-sentiment words, having information gain ratio above an optimal threshold, are included as features for creating the model. For example, in a DVD review, we found *“your money”* in the context of *”save your money and save your time”* has a negative connotation even though it is a non prior-polar phrase.

For creating an HAC, a combination of unigram, bigram and trigram features based on *TF-IDF* weighting scheme is used for representing data[1]. Unigram (F1), bigram(F2) and trigram(F3) features are concatenated to create the combined

[1] We choose a combination of unigram, bigram and trigram after conducting exhaustive experimentation. TF-IDF scheme was seen to work better compared to Term Presence or Term Frequency based feature representation.

model (F1+ F2+F3). As a pre-processing step, all review documents are converted to lower case. An IGR based feature selection is carried out to select the final features for creating the classifier. For finding the optimal information gain ratio threshold, we did tenfold cross validations over 1000 labeled data from each domain (from each category). A linear search is conducted by varying the information gain ratio threshold from 0 to 1 in steps of 0.01 for each set of ten-fold cross validations. The final threshold is selected from the run which gave the best average tenfold cross validation accuracy. Rapidminer 4.6 (Mierswa et al., 2006) with (LibSVM[2]) is used for experimentation. All other learning parameters are set to their default values. Linear kernel is used as it is fast and it yielded higher accuracy for most variations on the bag of words feature sets. Stemming and lemmatization are avoided as it is shown to be detrimental to classification accuracy (Leopold and Kindermann, 2002).

4.2 Intermediary Tagger System (ITS)

An HAC model cannot be directly used to accomplish CDST task. The sentiment prediction power of such a classifier would be limited when source and target domain are very different each from other. If an optimal amount of training data from target domain is obtained, an HAC on the target domain could be modeled. But HAC requires reasonable amount of *correct* training data for creating the model. In general, it is seen that *prediction power of many is better than one* (Aue and Gamon, 2005; Andreevskaia and Bergler, 2008). Thus, a cross domain sentiment prediction system based on committee of three classifiers is entrusted with creating this intermediate tagged data. We refer to it as **Intermediary Tagger System**.

A meta-classifier based approach is used for combining the individual predictions of the base classifiers. The first classifier, base 1 classifier, is modeled on domain pertinent information from source domain.

The second classifier, base 2 classifier, is modeled using generic features from a lexicon trained on the same domain as base 1 classifier. We refer to the lexicon[3] by Wilson *et al* (2005) as **Universal Sentiment Clues**. The lexicon consists of a set of manually identified 8000 words with their prior polarity. Apart from storing the prior polarity, the lexicon also indicates whether the word is strongly or weakly subjective. For example, *"wow"* is a frequently occurring word in product reviews. As per the lexicon, it is a *strongly subjective* word with *positive prior polarity*.

The third classifier, base 3 classifier, is based on SentiWordNet 1.0[4]. This Wordnet based resource has polarity scores attached to each senses (Esuli and Sebastiani, 2006). Each synset in this resource is marked with 3 scores- positive score, negative score and an objective score, with these scores summing to 1. The classifier considers the prior polarity of the words present in the document to calculate the overall positive and negative polarity score of the document.

We use *Meta-classifiers* or *stacking* as a way of combining classification models (Wolpert, 1992). The output of each model forms a new set of data. This along with the true label of the training instance becomes the input sample for the classifier in the next level. Wolpert (1992) called the original data and models constructed for it in the first step as *level-0* data and *level-0* model. The output of *level-0* is taken as the *level-1* data and the model created from that as the *level-1* model. The meta-classifiers learn how to combine the results of base classifiers and make the final prediction. In our case, the three classifiers form the *level-0* models and the *level-1* data needed for *level-1* model are prediction probabilities of positive and negative class of base 1 and base 2 classifiers and positive and negative sentiment scores of the document given by the base 3 classifier respectively.

The architecture of ITS is shown in figure 2. *Term Presence* feature[5] representation is used for both classifiers. Stemming of the corpus is done using the WordNet stemmer[6] for base 2 and base 3 classifier. A modified version of SVM which gives prediction probabilities along with labels are used for creating base 1 and base 2 classifiers (Wu et al., 2004). The learning parameters pertaining to all these models are kept to their default settings.

The SentiWordNet based classifier works on the

[2] http://www.csie.ntu.edu.tw/ cjlin/libsvm

[3] The lexicon can be downloaded from http://www.cs.pitt.edu/mpqa/lexiconrelease /collectinfo1.html.

[4] http://sentiwordnet.isti.cnr.it/

[5] Experiments with TF-IDF gave almost similar results.

[6] We used JWI API- http://projects.csail.mit.edu/jwi/

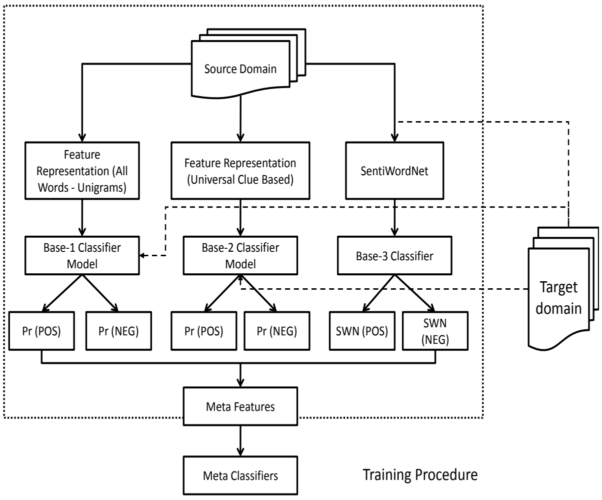

Figure 2: ITS Setup

following principle:

$$SWN(Neg) = \frac{\sum_{i \in W} \frac{\left(\sum_{s \in S} Neg(i_s) \right)}{S}}{W}$$

$$SWN(Pos) = \frac{\sum_{i \in W} \frac{\left(\sum_{s \in S} Pos(i_s) \right)}{S}}{W}$$

where

$$\begin{aligned} SWN() &= \text{Overall document polarity} \\ Neg()/Pos() &= \text{Neg/Pos score of synset} \\ W &= \text{number of words in a document} \\ S &= \text{number of senses for a word} \end{aligned}$$

```
if $SWN(Neg) < SWN(Pos)$ then
    Document is ***Positive***
else
    Document is ***Negative***
```

We refer the output of the base classifiers (*level-0* models) as meta-features. The meta-features comprise of prediction probabilities of base 1 and base 2 classifiers for positive and negative labels. The last pair of features consists of source document's positive and negative sentiment scores based on SentiWordNet.

To create *level-1* data provided by base 1 and base 2 classifiers for the source domain, the corpus is divided into ten equal sets each containing equal number of positive and negative instances and for each set, prediction probabilities are generated by creating different base 1 and base 2 classifier trained on the remaining nine sets. The process is then repeated on remaining nine sets to obtain the prediction probabilities of the entire training domain. We used SVM (nu-SVM with rbf kernel) provided by libSVM package for creating the level-1 model of the meta-classifier from the prediction probabilities and SWN scores. The class label is already known as the document belong to source domain.We did a "grid search" on γ and ν using tenfold cross validations. Various pairs of (γ, ν) values are tried and best cross-validation values are used.

As shown in the Figure 2, there are six meta-features (three pairs) for one training domain. The meta-features comprise of scores rather than labels which is one of the uniqueness of our system. The use of score increased the randomness of the features needed for creating better classification model. If we had followed the original meta-classifier model creation proposed by (Wolpert, 1992), number of possible combination of meta-features would have been limited to just 8. This would have lead to the creation of an inferior model.

For testing purpose, using the *level-0* models learned from source domain, prediction probabilities are generated for target domain dataset. The base 3 classifier is used directly on the target dataset to create individual document's positive and negative SWN scores. As mentioned before, these form the six meta-features of the target domain. The *level-1* model learned during the training phase is then applied on these meta-features for final prediction.

4.3 The Combination: H-I System

ITS does cross-domain sentiment prediction without using any label target data but with a lower accuracy whereas HAC requires reasonable amount of labeled data but would give us a model with high accuracy. The overall CDST accuracy can thus be bolstered by using HAC and ITS in tandem. We refer this system as **H-I** System.

ITS is used as an intermediary system to generate the tagged data for the HAC. The tagging done by ITS may be noisy. Noisy tagged data is filtered by retaining only *highly probable correct tagged target data* to create the HAC. By highly probable, we mean those instances which ITS has tagged with high confidence.The features selected, from wrongly tagged data, using the information gain ratio based selection process can hurt classification accuracy drastically.

Using different confidence level (labeling probability) provided by ITS, different set of intermediary tagged data $(T_1, T_2...T_n)$ (refer figure 1) are created. Different classification models $(M_1, M_2...M_2)$ are created from them and tested on small amount of hand-labeled target domain data. The model which gives the best in-domain accuracy, on the hand-labeled target domain data, is selected as the optimal model. The optimal intermediary tagged data selected using the approach mentioned minimize the amount of wrongly tagged data being included as training data(target domain).The training dataset corresponding to this model is used to train an HAC which completely tags the target domain with high accuracy.

Different sets of intermediary tagged data $(T_1, T_2...T_n)$ is generated by selecting document samples whose difference in positive and negative probability estimate is above a threshold. The threshold is varied from 0 to 1 with step size of 0.01. An SVM based on unigram *TF-IDF* feature vector representation is learned from each of these tagged data sets. An HAC is created using this intermediary tagged data after selecting an appropriate information gain ratio threshold as explained in section 4.1. The new target domain is then tagged using this model. We used 50 positive and 50 negative labeled samples[7] from the target domain for identifying the thresholds. ITS is trained on 1000 positive and 1000 negative samples from the training domain. We used 800 documents from each polarity set of the target domain for testing purpose.

Using the H-I system, we tackle the issue of unknown words indirectly. HAC requires domain pertinent features and these features are extracted from the intermediary target data. The in-domain classifier thus created has no or reduced effect of unknown words.

5 Dataset Used

We used Multi-Domain Sentiment dataset[8] used by Blitzer *et al* (2007) for our experiments. It contains corpora pertaining to 4 text domains - *DVD(DD), Electronics(ES), Kitchen(KN)* and *Books(BS)* taken from Amazon[9].

6 Results and Discussion

The result and discussion section is divided into three subsections for better clarity and comprehension.

6.1 HAC Results and Discussion

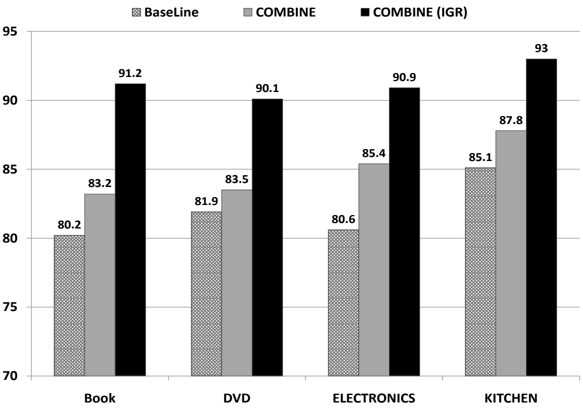

Figure 3: HAC performance comparison

The baseline for HAC is defined using a SVM based on unigram model[10]. We experimented with 3 different models based on the feature representation and IGR usage for label prediction. The results are shown in Figure 3. The best result is reported for the *combine*d of unigram, bigram and trigram with IGR. We observed an average increase of 9.35% in all the domains with respect to the baseline.

The considerable increase in the classification accuracy can be attributed to good features selected as part of IGR based feature selection. The final features selected for learning the model are highly discriminatory. Some of the top performing features of each domain are shown in the Table 2.

The top performing features contain both sentiment and non-sentiment bearing words. For example, "*horrible*" is a word with negative prior polarity. Feature like "*your money*" does not have any prior polarity but are highly discriminatory where the review contains phrases like "*save your money and save your time*" which gives the review on a DVD, a negative connotation. We found that a HAC trained on one domain performed with much

[7] We believe a single engineer can annotate at least 50 sample documents individually

[8] http://www.cs.jhu.edu/ mdredze/datasets/sentiment/

[9] www.amazon.com

[10] *TF-IDF* based representation is used

Domain	Top Information Gain Ratio based Features
BS	waste of, love this, boring, stupid, too many, whatever, ridiculous, two stars
DD	worst, horrible, your money, lame, of the best, sucks, barely, ridiculous, save your, is a great, pathetic, dumb, not worth, ruined
ES	return, terrible, waste your, highly, to return, poor, it back, returning, does not work, do not buy
KN	easy to, easy to use. easy to clean, returning, waste your, tried to excellent, defective, horrible, poor, i love it

Table 2: High information gain ratio based features

lesser accuracy when the target domain is different from training domain, which is expected as top information rich features are different in different domains. This is consistent with the observation made by Blitzer *et al* (2007). They showed that due to *domain specific* nature of sentiment analysis, *top information rich features are different in different domains*.

6.2 ITS Results and Discussion

To compare the results of the ITS, we define two baselines. The baseline 1 is defined by all-words base 1 classifier. The baseline 2 is defined by the universal sentiment clue based base 2 classifier. The intuition behind taking two baselines is that, the former enables a comparison of results with a model which is skewed to one domain as it is build on domain pertinent information and latter enables a comparison with a more generic model. A combination of these two classifiers along with SentiWordNet based classifier should give a better cross-domain classification accuracy compared to both baselines. The cross domain accuracy of ITS is compared in Figure 4. The in-domain classification accuracies of both classifiers are reported in the top line of the graph. B_1 stands for in-domain accuracy of base 1 classifier and B_2 for base 2 classifier. B_3 stands for accuracy provided by SentiWordNet. The objective of our approach is to beat in-domain as well as the best cross domain classification accuracy of both baselines. We tried out all the combination of classifiers to create ITS, but due to space constrains we have included the result of the best alone.

6.2.1 Effect of Domain Similarity on Baseline Results

After analyzing the classification accuracy of baselines, it is seen that for a given target domain some training domain is more suitable than other training domains. Moreover, this relationship is symmetric in our case, meaning that, when the training and the testing domains are interchanged, the observations still holds. For example, with *DVD* as the target domain, the model trained on the *Book* domain performs better than the other two available domains. The domains (*Electronics, Kitchen*) also showed the same phenomenon. Even when domains are interchanged, the observations still holds. The same phenomenon is observed for baseline 2 also. After analyzing these domains we felt there is a close lexical similarity between them. The cosine similarity[11] between the domains is summarized in Table 3.

Training/Target Domain	BS	DD	ES	KN
BS	1	**0.536**	0.410	0.399
DD	**0.536**	1	0.409	0.399
ES	0.410	0.409	1	**0.489**
KN	0.399	0.399	**0.489**	1

Table 3: Cross-domain cosine similarity

The *Book* and *DVD* domain are more similar compared to (*Book, Electronics*) or (*Book, Kitchen*). The same holds for *Electronics* and *Kitchen*. We observed, in general, baseline 2 accuracy is more than baseline 1 accuracy in a cross-domain setup which strengthens our belief of using universal sentiment clues in a general classifier.

6.2.2 ITS and Domain Similarity

The approach using meta-classifiers shows an improvement over baseline 1 and baseline 2. The cross-domain accuracy using models trained on different domains is also increased. From Figure 4, as expected, the prediction power of the group is more than the individual. Out of different training domains, similar domains have the best cross classification accuracy. All except *Electronics* → *DVD* and *Kitchen* → *DVD* have a cross classification accuracy more than the baselines. The baselines of these two pairs were low compared to the rest of the pairs. The similarity measure also suggested these two domains have wide disparity. The results of cross classification accuracy are observed to be high if the domains are similar. Since the meta-classifier learns how to select the final label based on the meta-features (*level-1*

[11] Term Presence feature representation is used

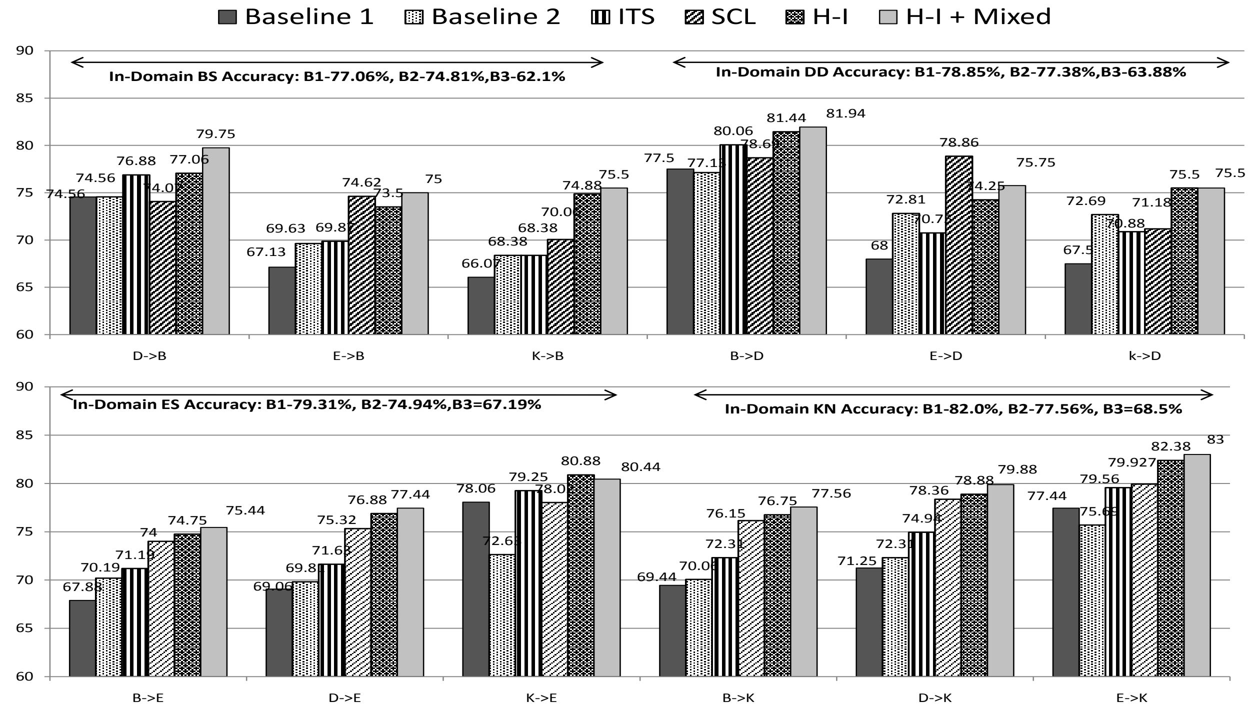

Figure 4: Classification accuracy comparison

data) it could get skewed to the training domain. As a result, if the domains are very dissimilar, it can be detrimental to prediction accuracy. Example, *DVD→Kitchen*.

6.3 H-I System Results and Discussion

The results of H-I system are reported in Figure 4. It is clearly evident from the figure that H-I performs much better than both the baselines. The domains which are similar as per Table 3 have tagging accuracy around 80%, which is expected because more amount of intermediary tagged data gets correctly tagged. Number of intermediary tagged instances selected from noisy tagged data after thresholding is shown in Figure 5. The graph explains the percentage accuracy achieved for optimal intermediary tagged target data. For each target domain, the optimal intermediary dataset created using a specific source domain and the percentage classification accuracy on the hand-labeled target data is shown. The number of training instances on different source domains are different because threshold for the selection of optimal intermediary target data using different ITS models are different.

Final H-I system is seen to have high accuracy on the domain where the intermediary tagged data has more number of correct instances. For example, from Figure 4 we observe that for

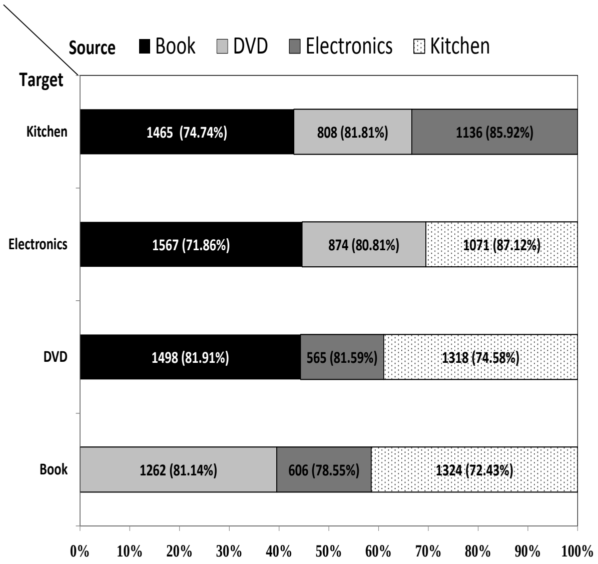

Figure 5: Selected optimal intermediary data and its accuracy on sample labeled target domain data

Kitchen as target domain the number of intermediary tagged instances selected after thresholding is maximum(1465) when *Book* is used as source domain. In statistical model based learning scenarios, availability of more training sample leads to a better classification model. But, the correctness of the training instances is more important than actual number of samples. As more correct instances are generated using *Electronics* as the source domain, H-I system produces best accuracy for *Kitchen* when *Electronics* is used as the source domain.

From our results, we note that increase in classification accuracy by using the H-I system is high for dissimilar domains than between similar domains. For example, with *Book* domain as the target domain, the percentage rise in tagging accuracy is higher for *Kitchen→Book* than *DVD→Book* with respect to baseline 1. The reason, we presume, for this being the lower baseline accuracies in these domains and because of high classification accuracy of HAC process, the relative rise in accuracy is more compared to a more similar domain. If the training and target domain are similar, ITS gives good classification results whereas if the domains are dissimilar, use of H-I system is a more appropriate choice.

6.4 Performance Comparison with SCL

In this section, we compare our performance with SCL[12], which is the commonly used algorithm for cross domain sentiment analysis. For creating HAC model for comparison, 50 hand-labeled samples from the target domain are mixed on top of the optimal intermediary tagged data selected. We refer this as **H-I + mixed** or in short, **H-I-M**. The same amount of data is mixed for judging the performance of SCL. The result from figure 4 shows a superlative performance of H-I-M over SCL system.

Table 4 summarizes all the results. The column *Best Baseline v/s H-I-M* and *Best H-I-M v/s SCL* show by how far the *Best H-I-M Accuracy* is better than the best baseline and the best SCL respectively. Using H-I-M process, the best CDST accuracies for all target domains is around 80%, which is well above in-domain accuracies of baseline classifiers. Moreover, CDST accuracy using H-I system gives an increment of 4.39% on an average over the baseline and 3.56% over SCL.

	Best Base Cross Domain Accuracy	Best H-I-M Accuracy	Best SCL Accuracy	Best Baseline v/s H-I-M	Best H-I-M v/s SCL
BS	74.56	79.75	74.07	5.19	5.68
DD	77.5	81.94	78.86	4.44	3.08
ES	78.06	80.44	78.02	2.38	2.42
KN	77.44	83.00	79.927	5.56	3.07
Average				4.39	3.56

Table 4: Result Summary and Comparison

7 Conclusion and Future Work

In this paper, we explained a promising approach for cross domain sentiment tagging. A method for creating high in-domain classifier using simple low level features is also introduced. A generic classifier based on meta-classification approach coupled with this high in-domain classifier is used in tandem to create labeled data for a new domain from domains having labeled data. Our results showed considerable improvement in cross domain sentiment tagging accuracy if domains are similar. More importantly, in case of dissimilar domains our system exceeds the baseline accuracies by substantial margins. H-I-M gives slightly better performance in comparison with commonly used SCL algorithm for cross domain sentiment prediction.

Quantification of the exact similarity threshold which can aid in choosing the training domain is a promising avenue for future research and we wish to pursue the same. Such a study can restrict the exhaustive prior search done for finding the best training domain for a target domain. We also plan to better the ITS to create more reliable intermediary tagged data which in turn can bolster the overall system performance.

References

Alina Andreevskaia and Sabine Bergler. 2008. When specialists and generalists work together: Overcoming domain dependence in sentiment tagging. In *Proceedings of ACL-08*, pages 290–298, Columbus, Ohio.

Anthony Aue and Michael Gamon. 2005. Customizing sentiment classifiers to new domains: A case study. In *Proceedings of RANLP-05*, Borovets, Bulgaria.

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of ACL-07*, pages 440–447, Prague, Czech Republic.

Sajib Dasgupta and Vincent Ng. 2009. Mine the easy, classify the hard: a semi-supervised approach to automatic sentiment classification. In *Proceedings of ACL-IJCNLP-09*, Morristown, NJ, USA.

Andrea Esuli and Fabrizio Sebastiani. 2006. SentiWordNet: A publicly available lexical resource for

[12] Implementation is available from author on request and parameters used for implementation is set to values mentioned in (Blitzer et al., 2007)

opinion mining. In *Proceedings of LREC-06*, Genova, Italy.

Edda Leopold and Jörg Kindermann. 2002. Text categorization with support vector machines. how to represent texts in input space? *Machine Learning*, 46(1-3):423–444.

Shotaro Matsumoto, Hiroya Takamura, and Manabu Okumura. 2005. Sentiment classification using word sub-sequences and dependency sub-trees. In *Proceeding of PAKDD-05*, volume 3518 of *Lecture Notes in Computer Science*, pages 301–310, Hanoi, Vietnam.

Ingo Mierswa, Michael Wurst, Ralf Klinkenberg, Martin Scholz, and Timm Euler. 2006. Yale: Rapid prototyping for complex data mining tasks. In *Proceedings of KDD-06*, pages 935–940, New York, NY, USA.

Tom M. Mitchell. 1997. *Machine Learning*. McGraw-Hill, New York.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, pages 1–135.

Jonathon Read. 2005. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of the ACL-05 Student Research Workshop*, pages 43–48, Morristown, NJ, USA.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of EMNLP-05*, pages 347–354, Vancouver, Canada.

David H. Wolpert. 1992. Stacked generalization. *Neural Networks*, 5(2):241–259.

Ting-Fan Wu, Chih-Jen Lin, and Ruby C. Weng. 2004. Probability estimates for multi-class classification by pairwise coupling. *J. Mach. Learn. Res.*, 5:975–1005.