www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 3 Issue 5 may, 2014 Page No. 5879-5885

Enhancing Accuracy in Cross-Domain Sentiment Classification by using Discounting Factor

K.Aarthi[1], C.S.Kanimozhi Selvi[2]

[1]PG Scholar, Department of Computer Science and Engineering, Kongu Engineering College,
Perundurai, Erode-638052, Tamilnadu, India
kannan.aarthi764@gmail.com

[2]Associate Professor, Department of Computer Science and Engineering, Kongu Engineering College,
Perundurai, Erode-638052, Tamilnadu, India
kanimozhi@kongu.ac.in

Abstract: ***Sentiment Analysis involves in building a system to collect and examine opinions about the product made in blog posts, comments, reviews or tweets. Automatic classification of sentiment is important for applications such as opinion mining, opinion summarization, contextual advertising and market analysis. Sentiment is expressed differently in different domains and it is costly to annotate data for each new domain. In Cross-Domain Sentiment Classification, the features or words that appear in the source domain do not always appear in the target domain. So a classifier trained on one domain might not perform well on a different domain because it fails to learn the sentiment of the unseen words. One solution to this issue is to use a thesaurus which groups different words that express the same sentiment. Hence, feature expansion is required to augment a feature vector with additional related features to reduce the mismatch between features. The proposed method creates a thesaurus that is sensitive to the sentiment of words expressed in different domains. It utilizes both labeled as well as unlabeled data of the source domains and unlabeled data of the target domain. It uses pointwise mutual information to compute relatedness measure which in turn used to create thesaurus. The pointwise mutual information is biased towards infrequent elements/features. So a discounting factor is multiplied to the pointwise mutual information to overcome this problem. Then the proposed method uses the created thesaurus to expand feature vectors. Using these extended vectors, a Lasso Regularized Logistic Regression based binary classifier is trained to classify sentiment of the reviews in target domain. It gives improved prediction accuracy than existing Cross-Domain Sentiment Classification system.***

Keywords: Sentiment Analysis**,** Cross-Domain, Classification, Lasso

1. Introduction

Data mining research has successfully produced numerous methods, tools, and algorithms for handling large amounts of data to solve real-world problems. Primary objectives of the data mining process are to effectively handle large-scale data, extract actionable patterns, and gain insightful knowledge. The explosion of social media has created extraordinary opportunities for citizens to publicly voice their opinions. Because social media is widely used for various purposes, vast amounts of user-generated data exist and can be made available for data mining.

Sentiment analysis analyzes people's opinions, appraisals, attitudes, and emotions toward entities, individuals, issues, events, topics, and their attributes. Users express their opinions about products or services they consume in blog posts, shopping sites, or review sites. Reviews on a wide variety of commodities are available on the Web such as, books (amazon.com), hotels (tripadvisor.com), movies (imdb.com), automobiles (caranddriver.com), and restaurants (yelp.com). It is useful for both the consumers as well as for the producers to know what general public think about a particular product or service. Sentiment analysis and opinion mining aim to automatically extract opinions expressed in the user-generated content. There are many social media sites reporting user opinions of products in many different formats. Monitoring these opinions related to a particular company or product on social media sites is a new challenge. Sentiment analysis and opinion mining tools allow businesses to understand product sentiments, brand perception, new product perception, and reputation management. These tools help users to perceive product opinions or sentiments on a global scale.

Sentiment analysis is hard because languages used to create contents are ambiguous. Performance evaluation of sentiment analysis is another challenge because of the lack of ground truth. Automatic document level sentiment classification is the task of classifying a given review with respect to the sentiment expressed by the author of the review. Sentiment classification has been applied in numerous tasks such as opinion mining, opinion summarization, contextual advertising and market analysis. For example, in an opinion summarization system, it is useful to first classify all reviews into positive or negative sentiments and then create a summary for each sentiment type for a particular product.

Supervised learning algorithms that require labeled data have been successfully used to build sentiment classifiers for a given domain. However, sentiment is expressed differently in different domains, and it is costly to annotate data for each new domain in which we would like to apply a sentiment classifier. A classifier trained on one domain might not perform well on a different domain because it fails to learn the sentiment of the unseen words. The Cross-Domain Sentiment Classification focuses on the challenge of training a classifier from one or more domains (source domains) and applying the trained classifier on a different domain (target domain). A Cross-Domain Sentiment Classification system must overcome two main challenges. First, it should identify which source domain features are related to which target domain features. Second, it requires a learning framework to incorporate the information regarding the relatedness of source and target domain features.

The rest of the paper is organized as follows: In section 2, related works are reviewed. Section 3 describes the concepts implemented in the proposed system. Section 4 describes the software and dataset used by the proposed system. Section 5 gives a brief overview analysis of results obtained. Section 6 concludes the report and identifies the scope for future work.

2. Literature Review

The automatic analysis of user generated contents such as online news, reviews, blogs and tweets can be extremely valuable for tasks such as mass opinion estimation, corporate reputation measurement, political orientation categorization, stock market prediction, customer preference and public opinion study. Also, new challenges raised by sentiment-aware applications are addressed [9]. Sentiment classification systems can be broadly categorized into single-domain and cross-domain classifiers based upon the domains from which they are trained on and subsequently applied to.

In single-domain sentiment classification, a classifier is trained using labeled data annotated from the domain in which it is applied. An investigation is done to determine whether it is sufficient to treat sentiment classification simply as a special case of topic-based categorization or whether special sentiment-categorization methods need to be developed [8]. This approach used three standard algorithms: Naive Bayes classification, maximum entropy classification, and Support Vector Machines (SVMs) for sentiment classification. In topic-based classification, all three classifiers have been reported to achieve accuracies of 90% and above for particular categories. This shows that sentiment categorization is more difficult than topic classification.

The co-occurrences between a word and a set of manually selected positive words (e.g., good, nice, excellent and so on) and negative words (e.g., bad, nasty, poor and so on) are measured using Pointwise Mutual Information (PMI) to compute the sentiment of a word [11]. An approach to build a domain-oriented sentiment lexicon is proposed to identify the words that express a particular sentiment in a given domain. By construction, a domain specific lexicon considers sentiment orientation of words in a particular domain. Therefore, this method cannot be readily applied to classify sentiment in a different domain.

A number of empirical tests on domain adaptation of sentiment classifiers are reported [1]. They used an ensemble of nine classifiers to train a sentiment classifier. However, most of these tests were unable to outperform a simple baseline classifier that is trained using all labeled data for all domains. They acknowledge the challenges involved in cross-domain sentiment classification and suggest the possibilities of using unlabeled data to improve performance.

The task of domain adaptation is investigated [3]. Empirical risk minimization offers better learning when training and test data come from the same domain. In the real world, people often wish to adapt a classifier from a source domain with a large amount of training data to different target domain with very little training data. This work proposed uniform convergence bounds for algorithms that minimize a convex combination of source and target empirical risk.

A semi-supervised (labeled data in source, and both labeled and unlabeled data in target) extension to a well-known supervised domain adaptation approach is proposed [6]. This semi-supervised approach to domain adaptation is extremely simple to implement, and can be applied as a pre-processing step to any supervised learner. However, despite their simplicity and empirical success, it is not theoretically apparent why these algorithms perform so well. Compared to single-domain sentiment classification, cross-domain sentiment classification has recently received attention with the advancement in the field of domain adaptation.

Sentiment is expressed differently in different domains, and annotating corpora for every possible domain of interest is impractical. It is necessary to train the classifier in multiple domains. Hence, the Cross-Domain Sentiment classification systems are reviewed. A Structural Correspondence Learning-Mutual Information (SCL-MI) algorithm is proposed for reducing the relative error due to adaptation between domains by an average of 30% over the original Structural Correspondence Learning (SCL) algorithm [2]. This work gives a new measure of domain similarity that correlates well with the potential for adaptation of a classifier from one domain to another. This measure of domain similarity can be used to select a small set of domains.

A general solution to sentiment classification is developed to address the cross-domain problem [7]. In this problem, the systems do not have any labels in a target domain but have some labeled data in a different domain, regarded as source domain. In this cross-domain sentiment classification setting, to bridge the gap between the domains, a Spectral Feature Alignment (SFA) algorithm is proposed to align domain-specific words from different domains into unified clusters, with the help of domain independent words as a bridge. In this way, the clusters can be used to reduce the gap between domain-specific words of the two domains, which can be used to train sentiment classifiers in the target domain accurately.

A fundamental problem when applying a sentiment classifier trained on a particular domain to classify reviews on a different domain is that words (hence features) that appear in the reviews in the target domain do not always appear in the trained model. To overcome this feature mismatch problem, a method called Sentiment Sensitive Thesaurus (SST) is proposed in cross-domain sentiment classification [4]. In this work, first a sentiment sensitive distributional thesaurus is created using labeled data for the source domains and unlabeled data for both source and target domains. Next, feature vectors are expanded using created thesaurus. Using these extended vectors, a Lasso regularized logistic regression based binary classifier is trained with source domain labeled reviews and is used to predict the sentiment of a target domain review.

PMI is commonly used to measure the association strength between two words. A problem of PMI is that it is biased towards infrequent elements/features. To overcome this problem a discounting factor is multiplied with PMI [10].

3. Enhanced Cross-Domain Sentiment Classification

A Cross-Domain sentiment classification system has been developed to overcome the problem of feature mismatch. It creates a sentiment sensitive thesaurus, using labeled data from source domains and unlabeled data from source and target domains. A problem of PMI is that it is biased towards infrequent elements/features. The relatedness between words in thesaurus is improved by multiplying a discounting factor with the pointwise mutual information. Then the proposed system extends feature vectors by using the created thesaurus. With the help of these extended vectors, a Lasso Regularized Logistic Regression based binary classifier is trained from the source domain labeled reviews to predict positive and negative sentiment in reviews. The Figure 1 depicts the working model of the proposed system.

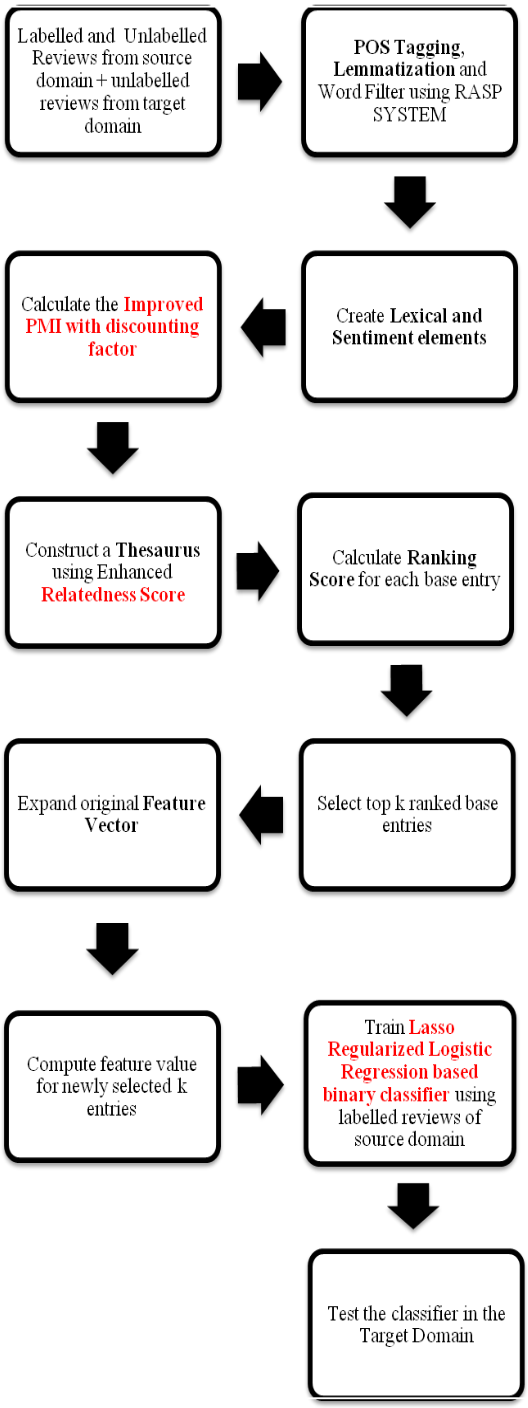

Figure 1: Working of proposed system

The following section gives a brief explanation of modules implemented for proposed system.

3.1 Generating Lexical and Sentiment Elements

The proposed system splits each review into individual sentences and then POS tagging and lemmatization is applied to these sentences using the RASP system. Lemmatization is the process of normalizing the inflected forms of a word (plural forms) to its lemma (singular form). Lemmatization reduces the feature sparseness and has shown to be effective in text classification tasks. Then a simple word filter based on POS tags is applied to filter out nouns, verbs, adjectives, and adverbs. In particular, adjectives have been identified as good indicators of sentiment. The proposed system models review as a bag of words and selects unigrams and bigrams from each sentence. The unigrams and bigrams are collectively as lexical elements. The lexical elements can be created from both source domain labeled reviews ($L(D_{src})$) as well as unlabelled reviews from source and target domains ($U(D_{src})$ and $U(D_{tar})$).

Next, from each source domain labeled review, sentiment elements are created by appending the label of the review to each lexical element generated from that review. The notation '*P' is appended to the lexical elements to indicate positive sentiment and '*N' is appended to the lexical elements to indicate negative sentiment. Sentiment elements, extracted only using labeled reviews in the source domain, encode the sentiment information for lexical elements extracted from source and target domains.

The Tables 1 illustrates the generation of lexical and sentiment elements for a review sentence in Electronics domain.

Table 1: Example for Electronics Domain

Steps	Example
A review sentence	I am happy to have several sandisk products and all of them are excellent
POS Tagging and Lemmatization	I//PPIS1 am//VBM happy//JJ to//TO have//VH0 several//DA2 sandisk//NN1 products//NN2 and//CC all//DB of//IO them//PPHO2 are//VBR excellent//JJ
Nouns, Verbs, Adjectives, and Adverbs	Happy, sandisk, products, excellent
Unigrams and Bigrams (Lexical Elements)	Happy, sandisk, products, excellent, happy+sandisk, snadisk+products, products+excellent
Sentiment Elements	Happy*p, sandisk*p, products*p, excellent*p, happy+sandisk*p, snadisk+products*p, products+excellent*p

3.2 Improved Pointwise Mutual Information

The proposed system represents a lexical or sentiment element *u* by a feature vector *u*, where each lexical or sentiment element *w* that co-occurs with *u* in a review sentence contributes a feature to *u*. From Table 1, for the lexical element 'Happy' in Electronics Domain, the other lexical and sentiment elements such as 'Happy, sandisk, products, excellent, happy+sandisk, snadisk+products, products+excellent, Happy*p, sandisk*p, products*p, excellent*p, happy+sandisk*p, snadisk+products*p, products+excellent*p' are considered as features. The value of a feature *w* in vector *u* is denoted by IPMI (u,w). IPMI (u,w) is the improved PMI value between a feature *w* and lexical or sentiment element *u*. IPMI is computed as follows:

IPMI (u,w) = f(u,w)* Discounting Factor(u,w)

$$f(u,w)=\log\left[\frac{\frac{c(u,w)}{N}}{\frac{\sum_{i=1}^{n} c(i,w)}{N} \times \frac{\sum_{j=1}^{m} c(u,j)}{N}}\right] \qquad (1)$$

Here, c (u, w) is the number of review sentences in which a lexical element *u* and a feature *w* co-occur, n is the total number of lexical elements, m is the total number of features and $N = \sum_{i=1}^{n}\sum_{j=1}^{m} c(i,j)$. f(u,w) is biased towards infrequent elements/features. So a discounting factor is multiplied with this f(u,w). The discounting factor is computed as follows:

Discounting Factor (u,w) =

$$\frac{c(u,w)}{c(u,w)+1} * \frac{\min\left(\sum_{i=1}^{n} c(i,w)\ \sum_{j=1}^{m} c(u,j)\right)}{\min\left(\sum_{i=1}^{n} c(i,w)\ \sum_{j=1}^{m} c(u,j)\right)+1} \qquad (2)$$

Therefore improved PMI can be computed using the following equation,

$$IPMI(u,w) = \log\left[\frac{\frac{c(u,w)}{N}}{\frac{\sum_{i=1}^{n} c(i,w)}{N} \times \frac{\sum_{j=1}^{m} c(u,j)}{N}}\right] * \frac{c(u,w)}{c(u,w)+1} * \frac{\min\left(\sum_{i=1}^{n} c(i,w)\ \sum_{j=1}^{m} c(u,j)\right)}{\min\left(\sum_{i=1}^{n} c(i,w)\ \sum_{j=1}^{m} c(u,j)\right)+1} \qquad (3)$$

3.3 Enhanced Relatedness Measure

Next, for two lexical or sentiment elements *u* and *v* (represented by feature vectors *u* and *v* respectively), the proposed system computes the relatedness T(v,u) of the element v to the element u as follows:

$$T(v,u)=\frac{\sum_{w \in \{x \mid IPMI(v,x)>0\}} IPMI(u,w)}{\sum_{w \in \{x \mid IPMI(u,x)>0\}} IPMI(u,w)} \qquad (4)$$

Here x is the features of the lexical elements. By using the relatedness measure defined in equation (4) a SST is constructed in which, for each lexical element *u*, the proposed system lists up lexical elements *v* that co-occur with *v* (i.e. IPMI(u, v) > 0) in the descending order of the relatedness values T(v,u).

3.4 Expanding Feature Vector

The proposed system performs feature expansion, where it augments a feature vector with additional related features selected from the sentiment-sensitive thesaurus. The system models a review d using the set $\{w_1,w_2,w_3,\ldots..w_N\}$, where the elements w_i are either unigrams or bigrams that appear in the review d. Then it represents a review d by a real-valued term-frequency vector d, where the value of the j-th element d_j is set to the total number of occurrences of the unigram or bigram w_j in the review d. The system defines a ranking score 'score (u_i,d)' for each base entry in the thesaurus as follows:

$$score(u_i,d) = \frac{\sum_{j=1}^{N} d_j T(w_j,u_i)}{\sum_{l=1}^{N} d_l} \qquad (5)$$

Here N represents the number of unigrams and bigrams appearing in the review, d is the review, u_i represents the base entry and w_j represents the words in the review d. The base entries of the thesaurus are listed in descending order of ranking score calculated with respect to unigrams and bigrams in the review. Then the system selects top *k* ranked base entries. Next, these top ranked *k* entries are appended to the review set d. The feature value of these newly appended base entries is calculated as follows: Let the *r*-th ranked ($1 \leq r \leq k$) base entry for a review d be denoted by v^r_d. The value of the *r*-th ranked base entry v^r_d is calculated as 1/*r*.

3.5 Lasso Regularized binary Classification

Generally, a learning algorithm is trained using some set of training examples. The learner is assumed to reach a state where it will also be able to predict the correct output for other examples, thus generalizing to situations not presented during training. However, especially in cases where learning was performed too long or where training examples are rare, the learner may adjust to very specific random features of the training data, which have no causal relation to the target function. This is the process of overfitting. In this process, the performance on the training examples still increases while the performance on unseen data becomes worse.

Logistic regression is a popular and well established classification method. Logistic regression is used for prediction of the probability of occurrence of an event by fitting data to a function. It is a generalized linear model used for binomial regression. When the number of feature variables is large compared to the number of training samples, logistic regression is prone to overfitting. To reduce overfitting, regularization is the best approach.

Using the extended vectors d' to represent reviews, the system trains a binary classifier from the source domain labeled reviews to predict positive and negative sentiment in reviews. Once an Lasso regularized logistic regression based binary classifier is trained, it can be used to predict the sentiment of a target domain review.

4. Experimental Setup

The proposed work can be carried out using Intel core i5 processor with 4GB RAM and 500GB Hard disk. The development environment used for implementing the proposed model is Python which runs on *Ubuntu 12.04*.3 *LTS 64-bit* operating system. The proposed system uses RASP system for POS tagging and lemmatization. The RASP system supports XML files for tagging. The system uses MLIB python Library for vector computation. The following section describes the RASP system and MLIB library used.

4.1 Software Used

4.1.1 Python 2.7

The Python 2 package contains the Python development environment. It is useful for object-oriented programming, writing scripts, prototyping large programs or developing entire applications. Python uses whitespace indentation, rather than curly braces or keywords, to delimit blocks. Python offers many built-in modules which will be useful for implementing Machine Learning Algorithms. As the proposed system will use Elastic-Net regularized logistic regression based Machine Learning Algorithm, python is selected as the development environment.

4.1.2 Robust Accurate Statistical Parsing

The Robust Accurate Statistical Parsing (RASP) [5] system (version 2) is used for POS tagging and Lemmatization process. This system was designed for syntactic annotation of free text. It has an enhanced grammar and POS tagger lexicon and a more flexible and semi-supervised training method for the structural parse ranking model.

The RASP system is designed to address several weaknesses of the RASP version1.First; all modules have been incrementally improved to cover a greater range of text types. Second, the POS tagger lexicon has been semi-automatically enhanced to better deal with rare or unseen behavior of known words. Thirdly, better facilities have been provided for user customization. The RASP system will be used by the proposed system for POS tagging and Lemmatization.

4.1.3 MLIB

MLIB is a collection of various machine learning algorithms and data mining algorithms. It was implemented in Python for research purposes by Danushka Bollegala, a Senior Lecturer in the University of Liverpool. MLIB is released under BSD License for non-commercial academic use. MLIB uses Python version 2.7 or later. The proposed system uses Truncated Gradient Descent classification algorithm of MLIB for the implementation of Lasso Classification.

4.2 Dataset Description

The proposed system uses the cross-domain sentiment classification dataset [2] to compare the proposed method against existing work on cross-domain sentiment classification. This dataset consists of Amazon product reviews for four different product types: books, DVDs, electronics and kitchen appliances. Each review is assigned with a rating (0-5 stars), a reviewer name and location, a product name, a review title and date, and the review text. Reviews with rating > 3 are labeled as positive, whereas those with rating < 3 are labeled as negative. The overall structure of this benchmark dataset is shown in Table 2.

Table 2: Number of reviews in Amazon Product Reviews Dataset

Domain	Positive	Negative	Unlabelled
Kitchen	800	800	16746
DVDs	800	800	34377
Electronic	800	800	13116

Books	800	800	5947

For each domain, there are 1000 positive and 1000 negative examples, the same balanced composition as the polarity dataset [8]. The dataset also contains some unlabeled reviews for the four domains. The proposed work randomly selects 100 positive and 100 negative labeled reviews from each domain as training instances (total number of training instances are 200*3 = 600). To conduct experiments, the proposed system selects each domain in turn as the target domain, with one or more other domains as sources.

5.3 Parameter for Evaluation

The classification Accuracy on target domain is used as a metric for evaluation. It is the fraction of the correctly classified target domain reviews from the total number of reviews in the target domain, and is defined as follows:

$$\text{Accuracy} = \frac{\text{Number of correctly classified target reviews}}{\text{Total number of reviews in the target domain}} \quad (5)$$

So the accuracy of prediction of target domain reviews by Elastic-Net regularized logistic regression based classifier is computed using equation (5). The accuracy measure gives the percentage of reviews that are correctly classified.

6 RESULTS

The proposed method improves the relatedness measure by multiplying a discounting factor with the existing PMI. This results in increase of relatedness measure. The figure 2 shows the increase in relatedness measure for the lexical element 'blackout'.

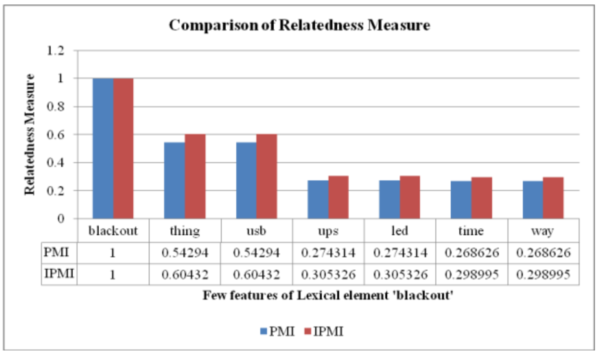

Figure 2 Comparison of Relatedness Measure

The proposed method performs feature expansion to overcome the problem of feature mismatch in Cross-Domains. The proposed method uses Amazon product review dataset which has four domains like electronics, dvds, kitchen and books. It takes one domain as target domain and the remaining three domains as sources domains. For example, if it takes dvds as target domain, the remaining three domains electronics, kitchen and books are considered as source domains. Similarly, if it takes books as target domain, the remaining three domains electronics, kitchen and dvds are considered as source domains.

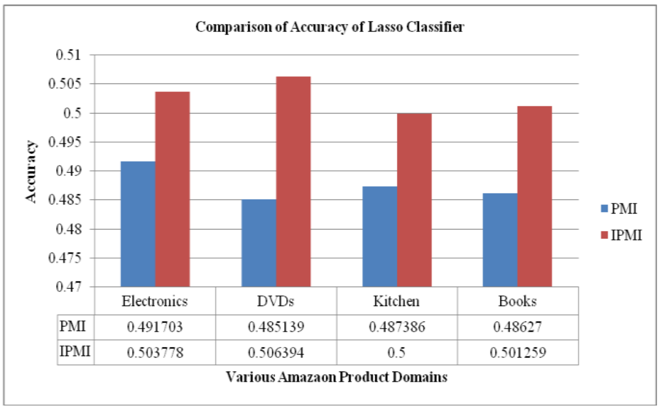

Figure 3 : Performance of Lasso Classifier for PMI and IPMI in various Domains

The proposed method performs feature expansion to overcome the problem of feature mismatch in Cross-Domains. The proposed method uses Amazon product review dataset which has four domains like electronics, dvds, kitchen and books. It takes one domain as target domain and the remaining three domains as sources domains. For example, if it takes dvds as target domain, the remaining three domains electronics, kitchen and books are considered as source domains. Similarly, if it takes books as target domain, the remaining three domains electronics, kitchen and dvds are considered as source domains. The figure 3 shows the performance analysis of Lasso classifier for PMI and IPMI in various domains. In electronics domain the lasso classification for IPMI gives 1% higher accuracy than PMI. Similarly, in dvds domain the lasso classification for IPMI gives 2% higher accuracy than PMI. Similarly, in kitchen domain the lasso classification for IPMI gives 1% higher accuracy than PMI. Similarly, in books domain the lasso classification for IPMI gives 2% higher accuracy than PMI.

The proposed method gives an average of 17.5% higher accuracy than Lasso and 14.5% higher accuracy than Ridge. And also the Elastic-Net Classification with IPMI gives an average of 7.25% higher accuracy than Elastic-Net Classification with PMI. Thus the proposed method performs better than the existing cross-domain sentiment classification system.

7 CONCLUSION AND FUTURE WORK

The proposed system develops a cross-domain sentiment classifier using an automatically extracted sentiment sensitive thesaurus. To overcome the feature mismatch problem in cross-domain sentiment classification, it uses labeled data from multiple source domains and unlabeled data from source and target domains to compute the improved relatedness of features and construct a SST. The proposed system extends the feature vectors by using the created thesaurus. Then using these extended vectors, a binary classifier is trained from the source domain labeled reviews to predict positive and negative sentiment in reviews. Once an Elastic Net regularized logistic regression based binary classifier is trained, it can be used to predict the sentiment of a target domain review.

The proposed method performs classification of only positive and negative reviews i.e., binary classification. In future, it can be extended to perform Multiclass classification of positive, negative and neutral reviews. It can also be extended to overcome the problem of word polymesy in cross-domains.

References

[1] Aue.A and Gamon.M (2005), ‘Customizing Sentiment Classifiers to New Domains: A Case Study’, Technical report, Microsoft Research.

[2] Blitzer.J, Dredze.M, and Pereira.F (2007), ‘Biographies, Bollywood, Boom-Boxes and Blenders: Domain Adaptation for Sentiment Classification’, Proc. 45th Ann. Meeting of the Assoc. Computational Linguistics (ACL ’07) pp. 440-447.

[3] Blitzer.J, Crammer.K, Kulesza.A, Pereira.F, and Wortman.J (2008), ‘Learning Bounds for Domain Adaptation’, Proc. Advances in Neural Information Processing Systems Conf. (NIPS ’08). pp. 1-8.

[4] Bollegala.D, Weir.D, Caroll.J (2013), ‘Cross-Domain Sentiment Classification using a Sentiment Sensitive Thesaurus’, IEEE Transcations on Knowledge and Data Engineering, Vol.25 , No.8 pp. 1719-1731.

[5] Briscoe.T, Carroll.J, and Watson.R (2006), ‘The Second Release of the RASP System’, Proc. COLING/ACL Interactive Presentation Sessions Conf. pp.77-80.

[6] Daum´e III.H, Abhishek.K, Avishek.S(2010), ‘Frustratingly Easy Semi-Supervised Domain Adaptation’, Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing, ACL 2010 pp. 53–59.

[7] Pan.S.J, Ni.X, Sun.J.T, Yang.Q, and Chen.Z (2010), ‘Cross-Domain Sentiment Classification via Spectral Feature Alignment’, Proc. 19th Int’l Conf. World Wide Web (WWW ’10).

[8] Pang.B, Lee.L, and Vaithyanathan.S (2002), ‘Thumbs Up? Sentiment Classification Using Machine Learning Techniques’, Proc. ACL-02 Conf. Empirical Methods in Natural Language Processing (EMNLP ’02) pp. 79-86.

[9] Pang.B and Lee.L (2008), ‘Opinion Mining and Sentiment Analysis’, Foundations and Trends in Information Retrieval, Vol. 2, No. 1/2 pp. 1-135.

[10] Pantel.P and Ravichandran.D (2004), ‘Automatically Labeling Semantic Classes’, Proc. Conf. North Am. Ch. Assoc. for Computational Linguistics: Human Language Technologies (NAACL-HLT ’04), pp. 321-328, 2004.

[11] Turney.P.D (2002) ‘Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews’, Proc. 40th Ann. Meeting on Assoc. for Computational Linguistics (ACL ’02) pp. 417-424.