Optimized Cross Domain Sentiment Classification Through N-Gram Features and Machine Learning

Chetan Sharma  
M. Tech. Scholar  
Department of Computer Science and Engineering  
Apex Institute of Engineering and Technology, Jaipur  
Rajasthan Technical University (Kota)

Dr. Bhavna Sharma  
Professor  
Department of Computer Science and Engineering  
Apex Institute of Engineering and Technology, Jaipur  
Rajasthan Technical University (Kota)

Abstract

World Wide Web is full of blogs and forums in which provides the users a platform to share their views or remarks about diversified topics. Sentiment analysis refers to the use of natural language processing, text analysis and computational linguistics to identify and extract subjective information in source materials. It aims to determine the attitude of a speaker or a writer with respect to some topic or the overall contextual polarity of a document. The attitude may be his or her judgment or evaluation, affective state or the emotional state of the author when writing, or the intended emotional communication the author wishes to have on the reader. Sentiment classification aims to automatically predict sentiment polarity (e.g., positive or negative) of users publishing sentiment data (e.g., reviews, blogs). Although traditional classification algorithms can be used to train sentiment classifiers from manually labeled text data, the labeling work can be time-consuming and expensive. Meanwhile, users often use different words when they express sentiment in different domains. Words in different domain can be treated as positive or negative and vice-versa, e.g. hard knife and hard pillow. Cross-domain sentiment classification can be done by using a spectral feature alignment (SFA) algorithm to align domain-specific words from different domains into unified clusters, with the help of domain independent words as a bridge. The proposed work extends the technique proposed by S J Pan et.al. by including character level N gram features and shorthand internet notations, usually used in the web, into sentiment classification. An SVM based classifier is proposed to classify the polarity of the reviews. Domain Independence is achieved using cross domain words. Compared to previous approaches, this technique can classify the documents with much more accuracy as the shorthand notations are increasingly popular among the internet users. Extensive experiments are performed on real world datasets of twitter and it is demonstrate that inculcation of N gram features and shorthand notations can provide much better results for polarity classification within smaller false positive and false negative rates.

Keywords: Sentiment Classification, Feature Extraction, Classification, Machine Learning etc.

I. INTRODUCTION

Sentiment Classification is different from traditional Information extraction in several ways. It includes comparatively few categories (positive/negative, 3 stars, etc) compared to text categorization. It usually comprises of crosses domains, topics, and users. The categories in this analysis are not independent (opposing or regression-like). Also, the characteristics of answers to opinion-based questions are different from fact-based questions. Thus, opinion-based Information Extraction (IE) differs from traditional IE. The main challenges in sentiment analysis are following:

1) People express opinions in complex ways
2) In opinion texts, lexical content alone can be misleading
3) Intra-textual and sub-sentential reversals, negation, topic change are common
4) Rhetorical devices/modes such as sarcasm, irony, implication, etc.

As an example of a sample document for opinion analysis, consider the following document:

Dear <hardware store>
Yesterday I had occasion to visit <your competitor>. They had an excellent selection, friendly and helpful salespeople, and the lowest prices in town. You guys cheat.
Sincerely,
<client>

This sample letter cannot be judged alone by textual analysis. It includes keywords like excellent, helpful salespersons, lowest prices etc., but is actually a negative document. Thus, other techniques needs to be augmented with sentiment classification so as to deduce the exact polarity of the document.

There are many possibilities for what can be classified. This includes:

1) Users
2) Texts
3) Sentences (paragraphs, chunks of text?)
4) Predetermined descriptive phrases (<ADJ N>, <N N>, <ADV ADJ>, etc)
5) Words
6) Tweets/updates

Classification of words and short phrases is one of the most important aspect of sentiment classification. These are building blocks of sentiment expression. Short phrases may be just as important as words: For example,

- lowest prices
- high quality

An approach is needed to deal with these before moving on to other classification tasks. This task of classification of words for positive and negative polarity is difficult and it depends on the linguistics of a particular person. For example:

| Table 1: Positive And Negative Word List |
|--------------------------------------|---------|---------|
| Proposed Word list                   | Accuracy| Ties    |
| **Human 1**                          |         |         |
| Pos: dazzling, brilliant, phenomenal, excellent, fantastic | 58%     | 75%     |
| Neg: suck, terrible, awful, unwatchable, hideous |         |         |
| **Human 2**                          |         |         |
| Pos: gripping, mesmerizing, riveting, spectacular, cool, awesome, thrilling, badass, excellent, moving, exciting | 64%     | 39%     |
| Neg: bad, cliched, sucks, boring, stupid, slow |         |         |
| **Statistics Based**                 |         |         |
| Pos: love, wonderful, best, great, superb, still, beautiful | 58%     | 58%     |
| Neg: bad, worst, stupid, waste, boring, ?! |         |         |

Data-driven methods can be used to generate keyword lists that model better than human-generated keyword lists. Unigram methods on similar data have reached 80% accuracy. One another approach for working with tweets and short text updates is the use of emoticons or similies. These are very little text to text with. Moreover, sentiment most succinctly represented with emoticons or smiley. Broadly, sentiment classification is the set of techniques that permits detection of emotional content in text. There are many applications related to it: it is normally used by trading algorithms to process news articles, as well as by corporations to better react to consumer service requirements. Techniques which are similar can also be applied to other text analysis problems, such as spam filtering.

Recently, the concern for “sentiment classification” has been gaining wide popularity. Bulky volumes of ratings, reviews, recommendations, and news corporations, online opinion might provide significant information for business to promote their product. Sentiment analysis also helps in keeping track of the mood of public about a specific product. Sentiment analysis also known as opinion mining involves building a system that collect opinions and examines them. This would be helpful in reviewing the success of a new product launched, in which manner new version of a product is received by the customer and its importance restricted to a certain area. The major concerns in sentiment analysis are an opinion word which is treated as indicating a positive side might be thought as negative in some other situation. Secondly, every person’s viewpoint is different. The conventional text processing takes into account that a little change in a couple of pieces of text will not change its meaning. But in sentiment analysis, the sentence “the banana is good” is different from “the banana is not good”. The majority of statement contains both positive plus negative opinions. The system processes it by examining each sentence one by one. Though, twitter and blogs consists of more casual statements that are simple to understand by the user. The customary text mining focuses on study of facts while opinion mining is related with the attitudes [1]. The major research fields are featuring based sentiment classification and opinion summarizing. Sentiment classification examines the opinions on a specific object. Feature based classification concentrates on classifying and analyzing depending on the features of the object [2].

As a promising communication platform, Web 2.0 makes the Internet to be progressively user interactive. People can convey and share their concerns and opinions in the cyberplace. Consequently, a number of user generated content consisting of great opinion and sentiment information has come out in the Internet. Perceiving such an opinion and sentiment information becomes progressively more significant for both service or product suppliers and users. Though, the opinion and sentiment classification is many a times either unstructured or semi-structured data over the Internet. For instance, online product reviews are time and again subjective, unstructured, and hard to digest in a short span of time. Sentiment classification aims to analyze direction-based text, such that text having opinions and emotions, to establish that whether a text is subjective or objective, and if a subjective text contains negative or positive sentiments. Sentiment classification techniques can be used to examine the sentiment information and the opinion in the Internet. Earlier studies show that two types of methods have been exploited for sentiment classification i.e., machine learning and semantic orientation. Every classifier is trained on a collection of representative feature in the machine learning approach. On the contrary, the semantic orientation approach does not necessitate prior training; in its place, it considers a word that how much it is liable to be positive or negative. Though, few studies have joint both the approaches for sentiment classification. In this study, lexicon enhanced method for sentiment classification is proposed which combines the two most important approaches for sentiment classification. Especially, the set of sentiment words collected using the semantic orientation approach as features in the approach of machine learning. These features are referred as “sentiment features” in this study. To express this proposed method, experimental studies were conducted using five sets of online product
analysis. Different feature sets consisting of content-specific, content-free, and sentiment features were also evaluated. Generally, sentiment analysis deals with the analysis of direction-based text. For example, text consisting of emotions and opinions. Sentiment classification research focuses on determining whether a text is subjective or objective, and whether a subjective text consists of negative or positive sentiments. The common two class issue concerns in classifying sentiments as either positive or negative. Further variations consist of classifying sentiments as opinionated or factual (Wiebe et al., 2001; Wiebe et al., 2004). A number of research attempts to classify emotions such as sadness, happiness, horror, anger, etc., in place of sentiments.

A. Sentiment Classification Methods
Two types of approaches are there in sentiment classification studies of machine learning

1) The Machine Learning Approach:
The machine learning approach consists of text classification techniques. This approach considers the sentiment classification problem as a topic-based text classification problem [3]. Any text classification algorithm can be used from the many which are available, e.g., SVM, naïve Bayes, etc. This approach was first tested by Pang et al. (2002) to categorize movie reviews in two classes i.e. positive and negative. This study performed the comparison between Maximum Entropy, naïve Bayes, and SVM. The uppermost classification accuracy of 83% was achieved by using SVM with 3-fold cross rationale.

2) The Semantic Orientation Approach:
The semantic orientation approach deals with classification based on negative and positive sentiment phrases and words enclosed in every assessment text. It does not need any former training so as to extract the data. Two types of techniques is employed in previous sentiment classification studies using the semantic orientation approach i.e. corpus-based techniques, and dictionary-based techniques.

3) Corpus-Based Techniques:
Corpus-based techniques attempt in finding the co-occurrence patterns of words to establish their sentiments. Different studies used different strategies to determine sentiments. Turney (2002) considered a phrase’s semantic point of reference to be the shared information among the phrase and the word “excellent as positive polarity subtracting the shared information among the phrase and the word “poor” as negative polarity. The polarity of whole text was estimated as the standard semantic orientation of the entire phrases that consists of adverbs or adjectives. Riloff and Wiebe (2003) [4] used a process of bootstrapping to study linguistically affluent patterns of subjective expressions so as to categorize subjective expressions from objective expressions. The process employed a pattern extraction algorithm to study possible subjective patterns beginning with a set of objective patterns taken from previous studies. The studied patterns were used to make a decision if an expression is subjective or not.

4) Dictionary-Based Techniques:
Dictionary-based techniques use antonyms, synonyms, and hierarchies in WordNet or added lexicons with sentiment information to establish word sentiments.

Each of the two approaches has its own advantages and disadvantages. The machine learning approach is supposed to be more precise than the semantic orientation approach. Though, a machine learning model is tuned to the training corpus, consequently, training is required if it is applied somewhere else. On the contrary, the semantic orientation approach has superior generality as compared to machine learning, but its classification accuracy is often not as great as compared to the machine learning approach. Chaovalit and Zhou (2005) [5] compared both these approaches on sentiment classification. They performed an experiment on movie reviews with two polarities- positive and negative. Their investigational results confirmed that the machine learning approach is far more exact but necessitate more time to train the model while the semantic orientation approach is less exact but is more proficient for real-time applications. They got the highest accuracy of 86% with 3-fold cross validation by using the machine learning approach and the maximum accuracy of 77% by using the semantic orientation approach. These outcomes are found to be even better than earlier findings.

B. Naïve Bayes Classifier
Naïve Bayes technique is a set of managed learning algorithms based on application of Bayes’ theorem with the “naïve” supposition of independence among each pair of features. Given a class variable $y$ and a dependent feature vector $x$ through $x_n$, Bayes’ theorem states the following relationship:

$$ P(y|x_1, \ldots, x_n) = \frac{P(y)P(x_1, \ldots, x_n|y)}{P(x_1, \ldots, x_n)} \quad \text{eq. 1} $$

With the use of the naïve independence assumption that

$$ P(x_i|y, x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) \quad \text{eq. 2} $$

for all $i$, this relationship is simplified to

$$ P(y|x_1, \ldots, x_n) = \frac{P(y) \prod_{i=1}^{n} P(x_i|y)}{P(x_1, \ldots, x_n)} \quad \text{eq. 3} $$

As $P(x_1, \ldots, x_n)$ is constant given the input, following classification rule can be used:

$$ P(y|x_1, \ldots, x_n) \propto P(y) \prod_{i=1}^{n} P(x_i|y) \quad \text{eq. 4} $$
\[ \hat{y} = \arg\max_y P(y) \prod_{i=1}^{n} P(x_i | y) \quad \text{eq. 5} \]

and Maximum A Posteriori (MAP) estimation is used to estimate \( P(y) \) and \( P(x_i | y) \): the previous one is the relative frequency of class \( y \) in the training set.

The dissimilar Naive Bayes classifiers varies mainly by the hypothesis they make concerning the distribution of \( (x_i | y) \).

Naive Bayes classifiers have performed well in many real world scenarios, beside their apparently easy hypothesis, prominently spam filtering and document classification. They need a small quantity of training data to approximate the essential parameters. Naive Bayes classifiers and learners can be very fast as compared to more complicated methods. The separation of the class conditional feature distributions specifies that every distribution can be separately estimated as a single dimensional distribution. This in order helps to ease problems arising from the bother of dimensionality.

### C. Max Entropy Classifier

Because of the least assumptions that the Maximum Entropy classifier crafts, it is used regularly when anything about the prior distributions is not known and when it is not valid to make any such hypothesis. Furthermore Maximum Entropy classifier is employed when the conditional independence of the features cannot be assumed. This is mainly in the case of Text Classification problems in which features are generally words which are clearly not independent. The Max Entropy needs more time to be trained as compared to Naive Bayes, largely due to the optimization problem that is required to be worked out so as to approximate the parameters of the model. However, after figuring out these parameters, the method present robust outcomes and it is competitive while considering memory and CPU consumption.

The main aim is to build a stochastic model, as described by Adam Berger (1996) [6], which exactly represents the performance of the random process in which the contextual information \( x \) of a document is taken as input and the output value \( y \) is produced. Considering the algorithm of Naive Bayes, the foremost step in building this model is to gather a large number of training data that consists of samples corresponding to the following format: \( x | y \), where the \( x \) contains the contextual information of the document which is the sparse array and \( y \) is considered as its class. The second step is to sum up the training sample representing the empirical probability distribution:

\[ \hat{p}(x, y) = \frac{1}{N} \times \text{number of times that } (x, y) \text{ occurs in the sample} \quad \text{eq. 6} \]

In which \( N \) is representing the size of the training dataset.

The over stated empirical probability distribution is used in order to build the statistical model of the random process which distributes texts to a specific class by considering their contextual information. The building blocks of this model will be the set of figures that is generated from the training dataset such as the empirical probability distribution.

The following indicator function is introduced:

\[ f_i(x, y) = \int_0^1 \begin{cases} 1 & \text{if } y = c_i \text{ and } x \text{ contains } w_k, \\ 0 & \text{otherwise} \end{cases} \]

The above indicator function is called as “feature”. This binary valued function returns 1 only in the case when the class of a specific document is \( c_i \) and the document hold the word \( w_k \).

Any statistic of the training dataset can be expressed as the predicted value of the suitable binary-valued indicator function \( f_i \). Therefore the estimated value of feature \( f_i \) regarding the empirical distribution \( \hat{p}(x, y) \) is equal to:

\[ \hat{p}(f_i) = \sum_{x,y} \hat{p}(x, y) f_i(x, y) \]

\[ \text{eq. 8} \]

Given that each training sample \( (x, y) \) occurs just once in training dataset then \( \hat{p}(x, y) \) is equivalent to \( 1/N \).

When a specific statistic is useful to this classification, the model is required to be in accordance with it. For doing so, the expected value is restricted that the model allocate to the predictable value of the feature function \( f_i \). The probable value of feature \( f_i \) regarding the model \( p(y|x) \) is equal to:

\[ p(f_i) = \sum_{x,y} p(x)p(y|x)f_i(x, y) \]

\[ \text{eq. 9} \]

Where \( p(x) \) is the empirical allocation of \( x \) in the training dataset which is generally set equivalent to \( 1/N \).

### D. Boosted Trees Classifier

Amongst such learning algorithms, algorithms based on boosting have several advantages such as: Boosting-based learning algorithms have been applied to Natural Language Processing problems effectively, together with text classification, zero-anaphora resolution, and so on. Moreover, classifiers that trained with boosting-based learners have provided faster speed of classification speed than Support Vector Machines with a tree kernel[7]. Though, presented boosting-based algorithms for semi-structured data, boosting algorithms for classification and for ranking, consists of various points for improvement. The weak learners which are employed in these algorithms learn classifiers which do not consider frequency of substructures. It happens as these algorithms take sentence as their input instead of a text or document having two or more sentences. Thus, even if vital
substructures appear a number of times in their target texts, these algorithms cannot match up such frequency. For instance, various types of negative expressions might be favored over a positive expression which comes several times in sentiment classification. Consequently, it might happen that a positive text which is using the same positive expression many times with some of negative expressions is considered as a negative text as frequency consideration is lacking.

E. Random Forest Classifier
The random forest is a collective approach that can also be considered as a type of nearby neighbor forecaster. Ensembles are considered to be a divide-and-conquer approach which is used to improve performance. The major belief behind ensemble methods is that a collection of “weak learners” can approach together to appear as a “strong learner”. Every classifier, independently, is a “weak learner,” whereas all the classifiers when combined are a “strong learner”. The data that needs to be modeled are the blue circles. It is assumed that they correspond to some primary function in addition to noise. Every independent learner is demonstrated as a gray curve. Each gray curve that denotes a weak learner is a good estimation to the primary data. The red curve (the ensemble “strong learner”) can though to be a much superior estimation to the underlying data.

II. PROBLEM STATEMENT, MOTIVATION AND RESEARCH APPROACH

In general, Humans are subjective creatures and opinions are important. Being able to interact with people at the level of sentiment classification has many advantages for information systems. Sentiment classification aims to automatically predict sentiment polarity (e.g., positive or negative) of users publishing sentiment data (e.g., reviews, blogs). In this dissertation, sentiment classification using cross domain words is extended by including the effect caused with the use of N gram character level representations and shorthand notations used on the web [2]. The real world data of twitter is analyzed and a SVM based classifier is trained on the data set. R statistical package is used for simulation. Results produced shows a considerable improvement over the base approach using only the cross domain words.

A good sentiment analysis technique is able to give the answer of the following general type of questions:

1) Is this product review positive or negative?
2) Is this customer email satisfied or dissatisfied?
3) Based on a sample of tweets, how are people responding to this ad campaign/product release/news item?
4) How have bloggers' attitudes about the president changed since the election?

Other related task that belong to the domain are:

1) Information extraction (subjective sentiment information).
2) Question answering (recognizing opinion-oriented questions).
3) Summarization (accounting for multiple viewpoints)
4) Flame detection, which refers to a hostile and insulting interaction between Internet users, often involving the use of profanity.
5) Identifying child-suitability of videos based on comments.
6) Bias identification in news sources
7) Identifying (in)appropriate content for ad placement

The business oriented decision making involves the answer of the following questions:

1) Question: Why aren’t consumers buying our laptop?
2) Search the web for opinions and reviews of this and competing laptops, Blogs, Epinions (a subject of public interest),
amazon reviews, tweets, etc.
3) create condensed versions or a digest of consensus points

Sentiment Analysis can also be helpful for cross domain applications like

1) Politics/political science
2) Law/policy making
3) Sociology
4) Psychology

Summarizing all the above mentioned points, one can state that sentiment analysis can enable numerous applications and possibilities. This includes analyzing of trends, identifying ideological bias, targeting advertising/messages, gauging reactions, Evaluation of public/voters' opinions etc.

Sentiment Analysis is also critical for a wide variety of other related disciplines. For example, Idea propagation through groups is an important concept in sociology (Rogers 1962, Diffusion of Innovations). Moreover, opinions and reactions to ideas are relevant to adoption of new ideas. Analyzing sentiment reactions on blogs [3] can give insight to this process through the modeling of trust and influence in the blogosphere using link polarity.

A. Research Approach
The polarity of a sentence is analyzed using the Natural language Processing Operations. Initially, two sets of reviews are taken, belonging to the category of positive and negative reviews, known as the test data set. Each set of reviews or the training set is first converted into a text corpus, or bag of words. This process is done using stop word removal and stemming. These words are
thereafter mapped to domain independent words using bipartite graph for cross domain analysis. This clustering is done using Spectral Feature Alignment technique. The SFA is further augmented with words belonging to shorthand notation and N grams. An SVM based binary classifier is then trained on the positive and negative examples which are composed of positive and negative reviews. The SVM is then tested on the real data set to classify the polarity. The proposed scheme produces a 5% improvement than that proposed by S. J Pan et.al. [4].

III. PROPOSED WORK

Cross Domain sentiment classification is one of the major areas of research in natural language processing as the review sites and blogs today comprises of diversified topics in various categories. For example, the twitter data consists of posts regarding cricket, bolly-wood, music, politics, current affairs and a lot more. Thus, a machine learning procedure for one specific domain may perform very poorly on some other domains. This is for the simple reason that the polarity of words in one domain may be different in some other domain. For example, a "slow music" phrase in song review might be positive but a "slow internet" in a mobile phone review is definitely negative. As a need of time, the sentiment classifier for modern web must be capable of cross domain sentiment classification which can only be achieved using mapping of words from cross domains to neutral words labeling the data to provide supervised learning. There exists a method to align domain-specific words from different domains into unified clusters, with the help of domain independent words as a bridge, which is called Spectral Feature Alignment, so as to map words in different domains using neural words. A simple model of sentiment classifier over a particular domain can be depicted as shown:

![Fig 1: Experimental Frameset](image)

B. Experiment Framework and Dataset

Since same words may have different polarity in different domains, a domain-independent sentiment word list is used to start the analysis. However, such a list is small initially and thus a self-growth algorithm is needed to bring up it. Considering the characteristics of the news, some features of the news documents are analyzed such as the length of the article, the polarity score of the sentiment words, the amount of the negation in one document, the polarity of the title, the first and the last sentence and so on. The features are about the structure or the content. In these features, the polarity of the title, first sentence, last sentence, the topic sentence and the whole document are all labeled by volunteers. The decision-tree method and SVM method is used to combine the features to generate sentiment classifiers. The classifiers are evaluated and results are compared.

C. Spectral Feature Alignment (SFA)

Spectral Feature Alignment (SFA) method is first proposed by Pan et al. [12]. In this technique, features are classified as domain-specific or domain-independent using the mutual information between a feature and a domain label. Both unigrams and bigrams are considered as features to represent a review. Next, a bipartite graph is constructed between domain specific and domain-independent features. Between a domain-specific and a domain independent feature in the graph an edge is formed if those two features co-occur in some feature vector. After that, spectral clustering is conducted to identify feature clusters. Finally, a binary classifier is trained using the feature clusters for classification of positive and negative sentiment. 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. The idea is that if two domain-specific words have connections to more common domain-independent words in the graph, they tend to be aligned together with higher probability. Similarly, if two domain-independent words have connections to more common domain-specific words in the graph, they tend to be aligned together with higher probability.
The specification of the problem can be done mathematically as follows:

A domain $D$ denotes a class of entities in the world or a semantic concept. For example, different types of products, such as books, DVDs and furniture, can be regarded as different domains. Take research area as another example, computer science, mathematics and physics can be also regarded as different domains.

Given a specific domain $D$, sentiment data are the text documents containing user opinions about entities of the domain. User sentiment may exist in the form of a sentence, paragraph or article. In either case, it corresponds with a sequence of words $w_1, w_2, w_x$, where $w_i$ is a word from a vocabulary $W$. In this work, the sentiment data is represented with a bag-of-words method, with $C(w_i, x_i)$ denotes the frequency of word $w_i$ in $x_i$ and the word sequential information is ignored.

Given a specific domain $D$, the sentiment data $x_i$ and $y_i$ denoting the polarity of $x_i$, $x_i$ is said to be positive if the overall sentiment expressed in $x_i$ is positive ($y_i = +1$), while $x_i$ is negative if the overall sentiment expressed in $x_i$ is negative ($y_i = -1$). A pair of sentiment text and its corresponding sentiment polarity $\{x_i, y_i\}$ is called the labeled sentiment data. If $x_i$ has no polarity assigned, it is unlabeled sentiment data. Besides positive and negative sentiment, there are also neutral and mixed sentiment data in practical applications. Mixed polarity means user sentiment is positive in some aspects but negative in other ones. Neutral polarity means that there is no sentiment expressed by users.

**D. Cross-domain Sentiment Classification using Character Level N-Grams**

Given two specific domains $D_{src}$ and $D_{tar}$, where they are referred to as a source domain and target domain respectively, suppose there is a set of labeled sentiment data $D_{src} = \{(x_{src}, y_{src})\}_{j=1}^{nsrc}$ in $D_{src}$, and some unlabeled sentiment data $D_{tar} = \{(x_{tar})\}_{j=1}^{narc}$ in $D_{tar}$. The task of cross-domain sentiment classification is to learn an accurate classifier to predict the polarity of unseen sentiment data from $D_{tar}$, using neutral or cross domain words and modern shortened notations using in chat and tweets. In order to solve this problem, a framework is proposed as shown in the figure 3.2.

![Fig. 2: Proposed Cross Domain Classifier](image)

There are primarily two fundamental tasks:

- To identify domain-independent features and
- To align domain-specific features.

In the first subtask, it is aimed to learn a feature selection function $\phi_{df}(\cdot)$ to select $l$ domain-independent features, which occur frequently and act similarly across domains $D_{src}$ and $D_{tar}$. These domain-independent features are used as a bridge to make knowledge transfer across domains possible. After identifying domain-independent features, $\phi_{df}(\cdot)$ can be used to denote a feature selection function for selecting domain-specific features, which can be defined as the complement of domain-independent features. In the second subtask, it is aimed to learn an alignment function $\varphi : R^{(m-1)} \rightarrow R^k$ to align domain specific features from both domains into $k$ predefined feature clusters $Z_1, Z_2, ..., Z_k$ s.t. the difference between domain specific features from different domains on the new representation constructed by the learned clusters can be dramatically reduced. For simplicity, $W_{q}$ and $W_{ds}$ is used to denote the vocabulary of domain-independent and domain-specific features respectively. Then sentiment data $x_i$ can be divided into two disjoint views. One view consists of features in $W_{df}$, and the other is composed of features in $W_{ds}$. $\phi_{df}(x_i)$ and $\phi_{ds}(x_i)$ is used to denote the two views respectively.

In the proposed scheme, the sentiment classifier is a function which can be defined as follows:

$$y = f(x) = \text{Polarity}(x \ast W^\top)$$

where
- $x$ = vector representation of the sentiment phrase from the vocabulary.
- $w$ = weight matrix of the words as learned by the classifier using labeled data.

where $\text{Polarity}(x \ast w^T) = +1$ if $x \ast w^T > 0$, and -1 otherwise.

IV. ANALYSIS OF PROPOSED WORK

A. Shorthand Internet Notations

A twitter-specific post that can contain up to 140 characters, images, or videos. Twitter is used largely for reporting real-time events, like sports, and sharing what one is doing. As of all internet linguistics, there are several shorthand notation used in twitter posts.

![Word-Cloud from the Corpus of Text from Positive Sentiments (Movie Reviews)](image1)

Fig. 3: Word-Cloud from the Corpus of Text from Positive Sentiments (Movie Reviews)

![Word-Cloud from the Corpus of Text from Negative Sentiments (Movie Reviews)](image2)

Fig. 4: Word-Cloud from the Corpus of Text from Negative Sentiments (Movie Reviews)
A framework is proposed to map the shorthand notations used now-a-days on the web portals and chat conversions, their actual words. This results in more accurate sentiment classification as compared to the one in which it is not present. The use of shortened notations in conversations is increasing at a considerable rate during past years, and thus it is inherently important to inculcate the same in the learning model for an accurate classification. Our experimental results on both document-level and sentence-level sentiment classification tasks demonstrate the effectiveness of our proposed framework. As a future prospective of this work, it is planned to encode some lexical knowledge from the source domain to the spectral domain-specific feature alignment framework. The reason is that each word has its polarity category. If we get the polarity knowledge of some words, one can adopt semi-supervised learning techniques to help learn more reasonable clusters of domain-specific features from the bipartite graph. In addition, it is planned to develop a more effective feature selection method to identify domain-independent features.

V. CONCLUSION AND FUTURE SCOPE
REFERENCES


