ISSN No. 0976-5697

Volume 8, No. 5, May – June 2017

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Dictionary based Sentiment Analysis of Hinglish text

Harpreet kaur
(Research Scholar)
UIET, Panjab University
Chandigarh, India

Dr. Veenu Mangat
(Assistant Professor)
UIET, Panjab University
Chandigarh, India

Nidhi
(Assistant Professor)
UIET, Panjab University
Chandigarh, India

Abstract: Sentiment analysis is a popular field of research in text mining. It involves extracting opinions from text such as reviews, news data, blog data etc. and then classifying them into positive, negative or neutral sentiments. Sentiment analysis of English has been explored but not much work has been done for Indian language. Some research has been carried in Hindi, Bengali, Marathi and Punjabi languages. Nowadays, a lot of communication in social media happens using Hinglish text which is a combination of two languages Hindi and English. Hinglish is a colloquial language which is very popular in India as people feel more comfortable speaking in their own language. This paper provides a survey of recent work done in the Hindi language along with a new proposed approach for sentiment analysis of Hinglish (Hindi + English) text.

Keywords: Sentiment Classification, Feature Extraction, Dictionary or Lexicon Development, Wordnet

I. INTRODUCTION

Sentiment analysis is a natural language processing task which is used to extract opinions or sentiments from the text for a given topic. Due to the rapid evolvement of social media, every person is digitally connected through social media. Now social media is not only used for chatting and file sharing, it has gone much beyond that. Many organizations use social media as a tool to understand the likes and dislikes of their customers [11]. This can be done through sentiment analysis or opinion mining. Sentiment analysis involves many tasks such as subjectivity detection, text preprocessing, feature extraction and sentiment classification. Some important notations are-

- Subjectivity/objectivity: Text which holds some sentiment is called subjective text. For example: "3 idiots bahut achi movie hai (3 idiots is an awesome movie)". On the other hand objective text does not hold any sentiment. For example- " Raj Kumar Hirani film ka director hai (Raj Kumar Hirani is the director of the movie)". For sentiment analysis, we only require subjective text which can be further classified into positive or negative. If we are taking objective text also, then we have to take three classes positive, negative or neutral.
- Polarity: Subjective text can be positive or negative. This is called polarity of text. Text can be of positive polarity or negative polarity.
- Sentiment level: Sentiment analysis can be performed at three levels-
  1) Document level in which the whole document is given positive or negative polarity [2].
  2) Sentence level in which each sentence is analyzed to give positive or negative polarity. Overall polarity is computed by counting the positive and negative comments. Majority comments decide the overall sentiment.
  3) Phrase level in which phrases or aspects in a sentence are analyzed to classify as positive or negative.

Most of the existing work has been done in English language But nowadays, in India, there is a lot of increase in Hinglish text. The Hinglish text is a combination of Hindi and English language. It is a Hindi sentence written using English language. For example: "Movie bahut achi thi (Movie was good)". This sentence has meaning in Hindi language but it is written using Roman English instead of Hindi Devanagri script. It is very important to analyze this type of text as most of Indians comment in this format only. A negligible amount of work has been done in this text. In Hindi also, little amount of work has been done. Hindi is resource scarce language [16]. This is also known as code-mix sentiment analysis.

II. SENTIMENT ANALYSIS PROCESS

Sentiment analysis process starts with *Data gathering* phase. Popular sources for sentiment analysis are tweets, movie reviews, product reviews, blog data, news data, etc. Then this text is preprocessed to remove some anomalies (eg. Stop word removal). *Text preprocessing* involves stop word removal and stemming. Stop words are the words in the text which do not contribute to any sentiment. For example: In "yeh movie bahut achi hai(This is a good movie)", 'yeh', 'hai',(this, is, a) are the stop words. Stemming is the process of removing prefixes or suffixes. For example: 'enjoying' or 'enjoyed' can be stemmed to 'enjoy'. Next step is to extract features from the text [3]. *Feature extraction* involves converting text dataset into feature vector or some other representations that make most important features of text available. Popular text feature extraction techniques are unigram, bigram or n-gram model, term frequency, POS tagging and tf-idf (term frequency inverse document frequency). Unigram feature set takes one word at a time. For example: “yeh movie bahut achi hai(This is a good movie)” will be taken as{yeh, movie,bahut, achi, hai (this, is, a, good, movie)}. In bigram, we take a pair of two and so on. For example {yeh movie. Movie bahut, bahut achi, achi hai(this is, is a, a good, good movie)}. Term frequency feature set takes into account the number of occurrences (or frequency) of a term. POS is Parts Of Speech tagging. As we know, adjectives (in the above example ‘good’) and adverbs contribute to most of the sentiment. So POS helps us in identifying adjectives and adverbs in a sentence. Tf-idf is the most informative set. It tells us how much a word is important in a document. It increases proportionally as the term frequency in a document increases but decreases if a term is occurring frequently in all the documents (document frequency). For example: stop words occurs in all the documents and do not facilitate any classification.

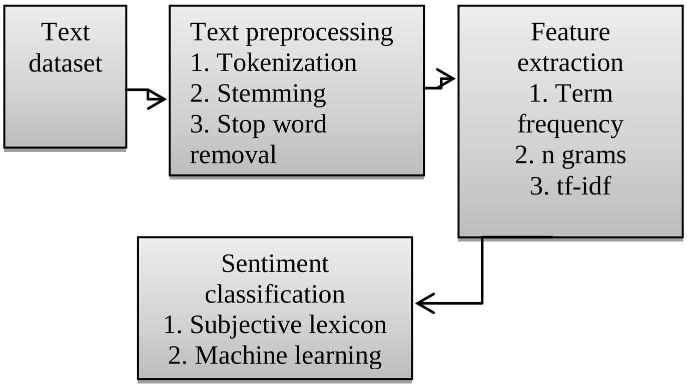


Figure 1. Sentiment analysis process

After feature extraction, the final phase is the *Sentiment classification*. The text is classified into positive and negative classes. There are mainly two approaches for this:

1) Subjective lexicon: In this approach, we have scores for each word indicating its positive, negative or neutral nature. For a given text, we sum all the positive, negative and neutral scores separately. In the end, highest score gives us the overall polarity. This approach can be further classified into dictionary based approach and corpus-based approach. Dictionary based approach involves the creation of seed list from opinion words from the dataset and then expanding it with the help of dictionaries or thesaurus. The corpus-based approach is also similar to dictionary based, the only difference is that in this we prepare seed list from the domain oriented corpus. For example, if our work is on movie reviews dataset, seed list will be prepared from movie domain text only. Corpus-based classification can be done in two ways- first is using a statistical technique which works on the basis of co-occurrence of words in the corpus i.e. if words occur mostly in the positive text then its polarity is positive otherwise negative. Another technique is Semantic-based approach. Wordnet is an example of this. It works on the principle of similarity between words. If some word in our dataset matches with the word in Wordnet, then we can use its score from SentiWordnet to find its polarity. [4]

2) Machine learning: It is an automatic classification process. Classification is performed using features which are extracted from the text(as explained above). It is of two types- Supervised and Unsupervised learning. Supervised learning involves training of classifier with labeled training data. Labeling means that class labels are known for each term in training dataset. Once the classifier is trained, it can be used to classify the testing data. On the other hand in unsupervised learning, no class label is known prior. The model makes inferences from incoming data and cluster it. Popular supervised learning classification algorithms are SVM(Support Vector Machine), NBC(Naïve Bayes classifier), ANN(Artificial Neural Network), RBFN(Radial Basis Function Network), ME(Maximum Entropy), etc.

SVM is the most popular classification algorithm. It creates a hyperplane which is used to classify data. It is a non-probabilistic classifier. If not properly trained, the problem of overfitting may arise i.e too much training leads to learning of noise in data as concepts to the classifier [1]. NBC is a probabilistic classifier which is based on Bayes ‘s theorem. It is easy to implement and can be updated easily. But its limitation is that it assumes that all the features are independent of each other. BN uses Directed acyclic graphs to represent dependencies between two variables. For example, a network depicting dependencies between symptoms and diseases can be used to find a disease for a given symptom. We can use ANN works similar as brain solves problems by neurons connected by axons. These are self-learned or self-trained systems. They require less data for training. But the system acts as a black box and we can’t view the relationships. RBFN is similar to ANN but it uses radial basis function as the activation function. It can be used for time series prediction. ME is a probabilistic classifier. It is used when we want to predict a dependent variable by using independent variables.

These are some popular classification algorithms used in sentiment classification. We are experimenting with these classification algorithms to find the best classifier results for Hinglish text.

III. LITERATURE SURVEY

Richa et al. [5] developed Hindi dictionary to find the polarity of Hindi reviews. They perform sentence-level sentiment analysis i.e., in the end, we get a total number of positive, negative and neutral sentences separately. The polarity of the sentence is determined by the presence of a majority of opinion words whether they are positive or negative. Movie domain dataset is first preprocessed and then it is given to Hindi POS tagger. This is used to identify adjective and adverbs which hold the most of the sentiment of the sentence. After this, a seed list is prepared with the most frequently used Hindi opinion words and their polarity scores. The output of POS tagger i.e. adjectives or adverbs are compared with this list. If a match is found then their scores are noted. If no match is found then their synonym is found and is matched with words in Hindi dictionary constructed by the authors. Then synonym is compared with seed list and if found then original word is added to seed list with same polarity score. Finally, the overall polarity is determined by the majority of words.

Kumar et al. [6] use Hinglish news and Facebook dataset to find the best pair of feature set and classifier algorithm. They conducted 840 experiments for this purpose. They conclude that term frequency-inverse document frequency (tf-idf) based feature representation, gain ratio based feature selection, and Radial Basis function(RBF) Neural Network are the best combination for sentiment classification. Training datasets are created by human annotators manually as no lexicon is available for Hinglish text. Text preprocessing involves tokenization, stop word removal; length based filtering of tokens, Document term matrix (DTM) preparation and feature selection. TF-IDF gives more information than term presence and term frequency. That is why it is used for feature extraction.

Pooja et al. [8] used Hindi SentiWordnet (HSWN) to find the sentiments related to Hindi movie review dataset. They performed document-level sentiment analysis. The polarity of words in the reviews is fetched from HSWN and then aggregated to find the overall polarity of the review. Negation handling is also performed. Words which are not present in HSWN, their polarity is found with the help of synset replacement algorithm. This system has two objectives: 1. Improving existing HSWN Wordnet. 2. Sentiment extraction. For the first objective, English SentiWordnet is used. Words which are not in HSWN are first translated into English and then their polarity scores are found. In the second objective, the sentiment is extracted by finding the overall polarity of the document that can be positive or negative or neutral.

Deepali et al. [11] develop an improvised polarity lexicon to overcome the limitation of existing HindiWordnet which is very generic in nature. They work on datasets of hotel and movie domain. The built lexicon reflects context sensitivity and shows an improvement in the accuracy. Their system has two objectives: To build an improvised context sensitive polarity lexicon for a specific domain and to improve the lexicon coverage by using the approach of HindiWordnet. Preprocessing involves tokenization, POS tagging, and lemmatization i.e. reduction to root word (For example 'lovely' or 'loved' can be reduced to 'love'). Then tf-idf score of each opinion word is calculated and words whose tf/idf scores are above some predefined threshold are kept and rest are discarded. Next step is to calculate final polarity score. This is termed as Context specific polarity lexicon (CSPL). Then adverbs and adjectives are extracted from this lexicon. Their synonyms are extracted from HindiWordnet. If a synonym is present in CSPL then there is no change in polarity score, otherwise extracted synonym is added to CSPL and same polarity score is assigned to it.

Shanshank et al. [9] perform sentiment analysis of code mix statements i.e. statements having both English and Hindi sentences. They use a statistical method to find the sentiment. In this, if a number of positive words in the statement is more, then the statement is termed as positive and vice versa. The methodology of this system involves language identification which involves tagging the words to corresponding languages as tag /E for English and /H for Hindi. After this spelling correction is performed e.g. happppyyy becomes happy. Then some words which are present in both languages (Hindi and English) like 'so', 'do', 'teen' etc. are handled in ambiguous word handling phase. Sounds like 'awww', 'boo', 'opps' are also taken into consideration. Phonetic variations are also handled. Then, Roman Hindi is transliterated to Devanagari Hindi. In the end, Wordnet is used for finding the polarity of English and HindiWordnet is used for finding the polarity of Hindi words.

Prashasti et al.[10] perform semantic analysis of Hinglish tweets. They use a dictionary based approach. For determining the polarity of tweets, a Scoring algorithm is used. Naïve Bayes classifier is used for classification purpose. They built two dictionaries one for positive words and other for negative words. The dictionaries contain all the words from tweets and also polarity scores associated with them. Words in tweets are matched with dictionary words.

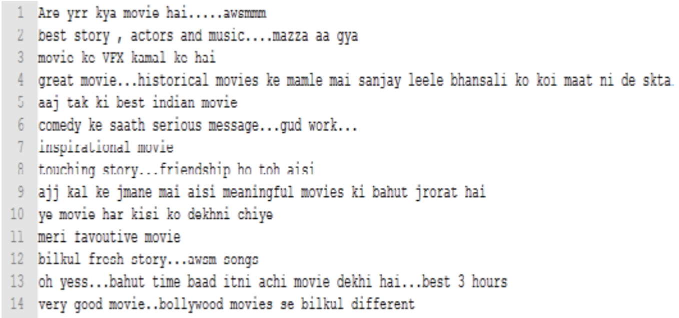
Shashank et al. [7] perform sentiment analysis of code mix data. They use FIRE 2014 and FIRE 2013 dataset which has data of English language mixed with six Indian languages (such as Tamil, Telugu, Hindi, Gujarati, Punjabi, Bengali). From that, they only use English and Hindi data. They also created their own corpus of 500 reviews from various sites. They use lexicon-based approach to find the sentiment of text. The first step in their model is to identify the language. If it is English, it is tagged as /E and if it is roman Hindi it is tagged as /H. Then /H tagged text is transliterated to Devanagari Hindi. Sentiment score of English and Hinglish text is calculated using Wordnet and HindiWordnet respectively. They have trained their model using three sentiment resources: opinion lexicon, AFINN(based on Affective Norms for English Words) and HindiSentiWordnet. Neutral English words are further classified using Wordnet. Their classifier achieves an accuracy of 85%.

Rupal et al. [13] perform multilingual sentiment analysis. The code mix consists of English and four other Indian languages (Tamil, Telugu, Hindi and Bengali). Multi-class SVM is used for classification. They use dictionary-based approach for training data for all the languages. They have taken FIRE 2015 dataset. Training dataset was built by crawling different websites like bengalilyrics.com, Tamil lyrics, and other Indian language websites. The first task is to identify the language and tag it. Indian languages are transliterated to their original form. N-gram features are extracted and classification is performed using multiclass SVM.

IV. PROPOSED APPROACH

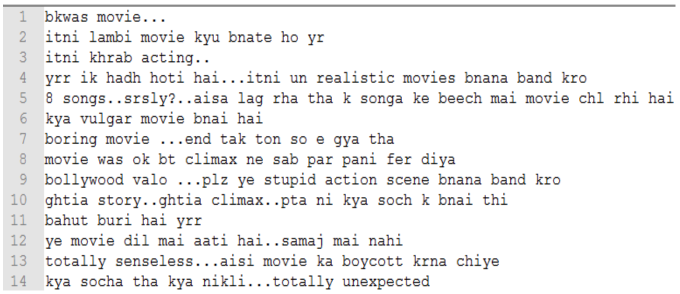
The survey of the literature shows that for Hindi and Hinglish text, most of the work has been done using translation. In code mix sentences (English + Hinglish), the first language of the word is identified. If it is English then it is tagged as /E and if Hindi then /H. After that sentiment polarity of English words is calculated with Wordnet. For Hindi text, they are first transliterated to Devanagari Hindi and then their sentiment polarity is calculated using HindiWordnet [5]. But this approach is not very effective as it involves translation from one language to another which is never 100% accurate. To solve this problem, we propose a dictionary based approach. We manually create a Hinglish dictionary which contains subjective (which holds some sentiment) words related to movie domain. We are working on movie reviews dataset. Hinglish reviews are manually collected from various social networking sites, blogs, YouTube etc. The methodology is broadly divided into following four sections:

4.1 Dataset collection: Sentiment analysis of Hinglish text is not a popular field. So there is not much data present. Some data sources like FIRE 2013, FIRE 2014 etc. are available but that also do not contain data for Hinglish sentiment analysis. They have data that is a mixture of five or six other Indian languages. We extracted data that is a mixture of Hindi and English only. Also, we manually collected data from various blogs, social networking sites like Facebook and twitter, YouTube etc. We have collected 100 positive and 100 negative comments related to movie domain. Some examples of the Hinglish dataset are shown in following diagrams.



```
Are yrr kya movie hai.....awsmmm
best story , actors and music....mazza aa gya
movic kc VFX kamal kc hai
great movie...historical movies ke mamle mai sanjay leele bhansali ko koi maat ni de skta
aaj tak ki best indian movie
comedy ke saath serious message...gud work...
inspirational movie
touching story...friendship ho toh aisi
ajj kal ke jmane mai aisi meaningful movies ki bahut jrorat hai
ye movie har kisi ko dekhni chiye
meri favoutive movie
bilkul fresh story...awsm songs
oh yess...bahut time baad itni achi movie dekhi hai...best 3 hours
very good movie..bollywood movies se bilkul different
```

Figure 2. Postive Hinglish movie reviews



```
bkwas movie...
itni lambi movie kyu bnate ho yr
itni khrab acting..
yrr ik hadh hoti hai...itni un realistic movies bnana band kro
8 songs..srsly?..aisa lag rha tha k songa ke beech mai movie chl rhi hai
kya vulgar movie bnai hai
boring movie ...end tak ton so e gya tha
movie was ok bt climax ne sab par pani fer diya
bollywood valo ...plz ye stupid action scene bnana band kro
ghtia story..ghtia climax..pta ni kya soch k bnai thi
bahut buri hai yrr
ye movie dil mai aati hai..samaj mai nahi
totally senseless...aisi movie ka boycott krna chiye
kya socha tha kya nikli...totally unexpected
```

Figure 3. Negative Hinglish movie reviews

4.2 Dictionary Building Phase: As mentioned above, no Hinglish corpus is available for sentiment analysis of the Hinglish text. Wordnet and HindiWordnet are available for English and Hindi language respectively. These are lexicon sources which contain words related to English and Hindi. SentiWordnet and HindiSentiWordnet are also developed to find the polarity scores for words. For example: consider a review like "Movie is good". Now in order to find whether this review is positive or negative, look up the score of word 'good' in SentiWordnet. In SentiWordnet, each word has two scores associated with it in the range of 0 to 1 which tells us whether the word is positive or negative. Similarly, for Hindi words, scores can be calculated using HindiSentiWordnet. But for Hinglish text, no such lexicon is available. So our aim is to build a Hinglish dictionary or lexicon for movie domain.

Table 1. Extraction of opinion words from dataset

Movie reviews	Opinion words
best story, actors, and music....maza aa gya	Best, maza
movie ka screenplay bahut acha hai	Acha

In the above example, first, we extract adjective words from the text.[15] Then we create two separate dictionaries-one for English data (which will contain English opinion words like 'best' in above example) and one for Hinglish data ('mazza', 'acha'). In dictionary creation, we first extract opinion words from the dataset as shown in table 2. Then we place Hinglish and English opinion words in separate files. Next step is to populate our dictionaries by using synonyms of these words so as to maximize the coverage as shown in table 3. English words are populated using their synonyms from Wordnet. Similarly, Hinglish words are populated by using their Hindi equivalent in HindiWordnet.

Table 2. Populating lexicon with synonyms of opinion words

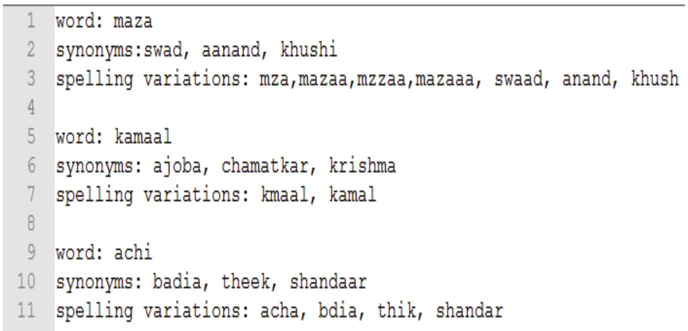
Opinion word	Synonyms	Source used
Best	Fine, great, excellent	Wordnet
Maza	swad, aanand, khushi	HindiWordnet
Acha	badia, theek, shandaar	HindiWordnet

Additionally, there are many variations of opinion words are usually found in reviews. To handle these variations, we add some variations to our dictionary as shown in table 4. Our system will not cover all possible variations but the most frequently used will be covered. Also, these variations are case-insensitive i.e. whether it is 'acha' or 'Acha', they will be considered as same.

Table 3. Taking into account the spelling variations

Opinion word	Popular variations
Best	Besttt, bstttt
Maza	Mazza, mazaaa
Acha	Achaa, achi, achaa

Now to calculate polarity scores of Hinglish words(eg. Maza, acha), we are taking help from HindiSentiWordnet as they have meaning in Hindi language only. For example: to calculate polarity score of 'maza(enjoyment)' we will find its Hindi equivalent in HindiSentiWordnet and assign it the same score. Some words from the dictionary are presented below:



```
word: maza
synonyms:swad, aanand, khushi
spelling variations: mza,mazaa,mzzaa,mazaaa, swaad, anand, khush

word: kamaal
synonyms: ajoba, chamatkar, krishma
spelling variations: kmaal, kamal

word: achi
synonyms: badia, theek, shandaar
spelling variations: acha, bdia, thik, shandar
```

Figure 4. Hinglish dictionary

Hinglish stop word list preparation: Next step is to create stop word list for Hinglish dataset. Stop words do not contribute to any sentiment. Stop word list for English text is available online. For Hinglish text, it is manually created by using Hindi stop word list available online as shown in table 5.

Table 4. Preparation of stop word list for Hinglish text

Review	Opinion words	Stop words
best story, actors, and music....maza aa gya	Best, maza	And, aa, gya
movie ka screenplay bahut acha hai	Bahut acha	Ka, hai

V. METHODOLOGY

Up to this step, our dictionary building module is done. Next step is to use these dictionaries for sentiment analysis as follows-

5.1 Text Preprocessing: It involves following tasks:

5.1.1 Tokenization: It is the process of converting a text stream into small individual tokens.

5.1.2 Stop word removal: Stop words do not contribute to any sentiment. Therefore there removal is required as they lead to decrease in accuracy. English stop word list is available online. For Hinglish, we will use our own created Hinglish stop word list [14].

5.1.3 Spelling variation checking: Many variations of opinion words are found in reviews. To handle these variations, spelling variation lists will be used which are created in Section 4.

5.2 Feature Extraction: It is the process of converting the text into some representations that make the important text available for analysis. Extracted features are then used to train our classifier. According to our literature survey, tf-idf (term frequency–inverse document frequency) feature set yields the best results for Hinglish text (using transliteration) [8]. As we are using dictionary based approach (without transliteration). We are also experimenting with other feature sets such as unigram, bigram, trigram along with tf-idf. Negation handling is also done in this.

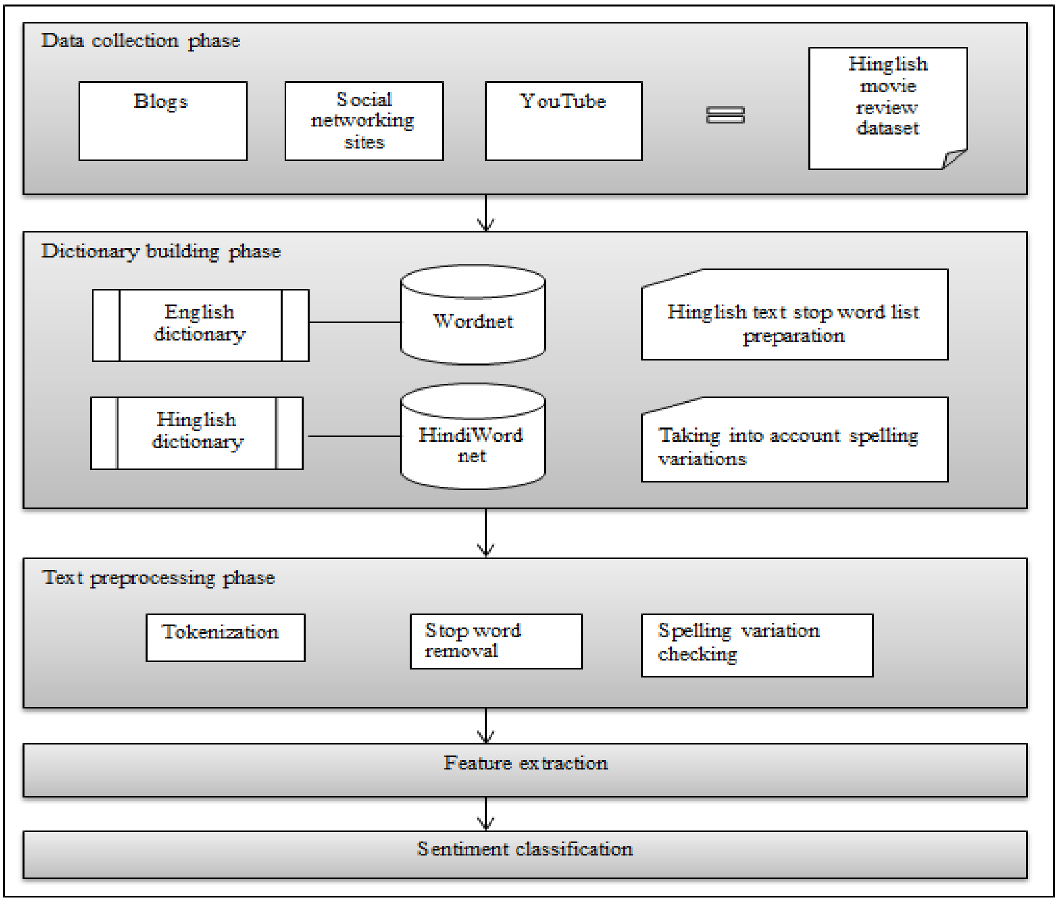


Figure 5. Proposed system for Hinglish text sentiment classification

5.2.1 Tf-idf: It stands for Term Frequency–Inverse Document Frequency. This is a feature extraction technique which is used to find the importance of a word to a document in a corpus. It increases as the number of occurrence of word increases in a document and decreases as the number of occurrences of the word increases the corpus. To compute it, we need two things: term frequency and inverse document frequency.

Term frequency: It tells us the occurrences of the word in a document. It can be computed as

$$\mathrm{Tf} = (\text{Occurrences of term in document} \,/\, \text{Total number of terms in document})$$

Inverse document frequency: It tells us how important a term is. While calculating tf, all terms are considered as important but we know that some terms like ‘is’, ‘are’, ‘the’ are frequent in every document and thus are less important. It can be calculated as

$$\mathrm{Idf} = \log\_e(\text{Total number of documents} \,/\, \text{Number of documents with term t in it})$$

Then tf/idf can be calculated by multiplying these two.

$$\mathrm{Tf/idf} = \mathrm{tf} * \mathrm{idf}$$

5.2.2 Negation handling: Negation means that review contains words like not, don’t etc. These words reverse the overall sentiment of the review. For example: “Movie is not good”. Since it contains the word ‘good’, the classifier can think that it is a positive review. Similarly, in Hinglish, we have this comment as “Movie achi nahi hai”. To handle this kind of comments, immediate previous and immediate next word to opinion word (e.g. ‘Achi’ or ‘good’) are checked. If any one of them is nahi (not) or any other negation word(e.g. don’t, didn’t), then their sentiment scores are reversed. [12]

5.3 Sentiment Classification: After feature extraction, we get feature scores which are used to train the classifier. We have taken a number of classifiers such as SVM, Naïve Bayes, Neural network, Logistic regression etc for experiments. Our goal is to find the best feature set and classifier for Hinglish text.

<u>For example</u>: To understand the overall procedure of Hinglish sentiment analysis, consider a movie review “ yr kya movie hai….mazza aa gya…thoroughly enjoyed it for 3 hours….music bhi kmaal ka hai….everyone must see it ”

Step 1: Text Preprocessing, it involves-

1.1 Tokenization: It is the process of token creation of a text stream.

‘yr’, ‘kya’, ‘movie’, ‘hai’, ‘mazza’, ‘aa’, ‘gya’, ‘thoroughly’, ‘enjoyed’, ‘it’, ‘for’, ‘3’, ‘hours’, ‘music’, ‘bhi’, ‘kmaal’, ‘ka’, ‘hai’, ‘everyone’, ‘must’, ‘see’, ‘it’

1.2 Stop word removal: Removing words which do not contribute to polarity detection

‘kya’, ‘movie’, ‘mazza’, ‘thoroughly’, ‘enjoyed’, ‘music’, ‘kmaal’, ‘must’, ‘see’

1.3 Spelling variations: During dictionary matching, it is found that ‘mazza’ is same as ‘maza’. So we will use the score of ‘maza’ for ‘mazza’.

Step 2: Feature Extraction.

2.1 Tf/idf score of each word is calculated and a threshold is set for noise removal. The words whose score is below the threshold can be ignored for further analysis. Let us assume that terms after threshold checking are

‘mazza’, ‘enjoyed’, ‘kmaal’

2.2 Negation handling in this example is not required as there are no negation words e.g. ‘nahi’ (not).

Step 3: Sentiment Classification.

The classifier will check these terms. As it contains most of the positive words(‘mazza’, ‘enjoyed’, ‘kmaal’), this review will be given a label of a positive review and it will belong to the positive class.

VI. CONCLUSION

A little amount of work has been done for Hinglish text using translation which is never 100% accurate and leads to decrease in accuracy. In this paper, we propose an approach for sentiment analysis of the Hinglish text by using dictionary based technique. Hinglish movie review dataset has been collected from various sources. In dictionary building module, we prepare two dictionaries: one for English data and another for Hindi data which are capable of handling word variations and are case-insensitive. Hinglish stop word list is also prepared. For feature extraction, tf-idf technique is used along with unigram, bigram and trigram. Negations handling is also performed. For sentiment classification, we are experimenting with popular classification algorithms. Our goal is to find the best feature extraction and classification technique for Hinglish text sentiment analysis.

REFERENCES

[1]. Pang, B., Lillian L., and Shivakumar V. Thumbs up?: Sentiment Classification Using Machine Learning Techniques. Proceedings of the ACL-02 Conference on Empirical methods in Natural Language Processing, 10. Association for Computational Linguistics, (2002).

[2]. Moraes, R., Valiati, J. and Gaviao Neto, W. Document-level sentiment classification: An empirical comparison between SVM and ANN. Expert Systems with Applications 40, 2(2013), 621-633.

[3]. Kaur, A. and Gupta, V. A Survey on Sentiment Analysis and Opinion Mining Techniques. Journal of Emerging Technologies in Web Intelligence 5, 4 (2013). DOI:10.4304/jetwi.5.4.367-371

[4]. Medhat, W., Hassan, A. and Korashy, H. Sentiment analysis algorithms and applications: A survey. Ain Shams Engineering Journal 5, 4 (2014), 1093-1113

[5]. Sharma, R., Nigam, S., and Jain, R. Polarity Detection of Movie Reviews in Hindi Language. International Journal on Computational Science & Applications 4, 4 (2014), 49-57

[6]. Ravi, K. and Ravi, V. Sentiment classification of Hinglish text. 3rd IEEE International Conference on Recent Advances in Information Technology (RAIT) (2016).

[7]. Sharma, S., P. Y. K. L., S. and Rakesh Chandra, B. Text normalization of code mix and sentiment analysis. IEEE International Conference on Advances in Computing, Communications and Informatics (2015).

[8]. Pandey, P. and Govilkar, S. A Framework for Sentiment Analysis in Hindi using HSWN. International Journal of Computer Applications 119, 19 (2015), 23-26.

[9]. Sharma, S., P. Y. K. L., S. and Rakesh Chandra, B. Sentiment analysis of code-mix script. IEEE International Conference on Computing and Network Communications (2015).

[10]. Kanikar, P., Koppisetty, R., Govindan, S., Bhat, S. and Virani, M. Semantic Analysis on Twitter Data Generated by Indian Users. International journal of scientific and engineering research 7, 9, (2016).

[11]. Mishra, D., Venugopalan, M. and Gupta, D. Context Specific Lexicon for Hindi Reviews. Procedia Computer Science 93, (2016), 554-563.

[12]. N. Mittal and B. Agarwal, “Sentiment Analysis of Hindi Review based on Negation and Discourse Relation”, International Joint Conference on Natural Language Processing, 2013, 45-50.

[13]. Bhargava, R., Sharma, Y. and Sharma, S. Sentiment analysis for mixed script Indic sentences. IEEE International Conference on Advances in Computing, Communications, and Informatics

[14]. Kanikar, P., Koppisetty, R., Govindan, S., Bhat, S. and Virani, M. Semantic Analysis on Twitter Data Generated by Indian Users. International journal of scientific and engineering research 7, 9, (2016).

[15]. P. arora, A. bakliewal and V. verma, "Hindi Subjective Lexicon Generation using Wordnet graph traversal", International Journal of Computational Linguistics and Applications, 3, 1, (2012) .

[16]. R. Sharma, S. Nigam and R. Jain, "Opinion Mining In Hindi Language: A Survey", International Journal in Foundations of Computer Science & Technology, 4, 2, (2014), 41-47.