Sentiment Analysis for Bangla Microblog Posts

Thesis Report submitted in partial fulfillment of the requirement for the degree of

Bachelor of Science in Computer Science and Engineering

By

Shaika Chowdhury

Student ID: 10101037

shaika.chy@gmail.com

Wasifa Chowdhury

Student ID: 10101038

wasifa.chy@gmail.com

BRAC University, Dhaka, Bangladesh

Fall 2013

DECLARATION

We hereby declare that this thesis is based on the results found by us. Materials of work found by other researchers are mentioned by reference. This thesis, neither in whole nor in part, has been previously submitted for any degree.

_______________________ _______________________

Signature of Supervisor Signatures of Authors

ACKNOWLEDGMENTS

We want to express our gratitude to The Almighty for blessing us with patience and knowledge and giving us the opportunity to learn something new. We would also like to thank our parents and teachers for their support and encouragement.

TABLE OF CONTENTS

DECLARATION......2

ACKNOWLEDGMENTS......3

LIST OF TABLES AND FIGURES...... 5

ABSTRACT......6

2. RELATED WORKS IN ENGLISH......9

3. LEXICON......11

3.1 METHODOLOGY TO CONSTRUCT THE BANGLA SENTIMENT LEXICON:...... 11

4. METHODOLOGY......13

4.1 DATASET:...... 13
4.2 TWEET DESCRIPTION:...... 14
4.3 PREPROCESSING:...... 14
4.4 OUR APPROACH...... 17
4.4.1 TRAINING SET CONSTRUCTION:...... 17
4.4.1.1 SEMI-SUPERVISED:...... 17
4.4.1.2 SELF-TRAINING BOOTSTRAPPING:...... 17
4.4.2 FEATURE EXTRACTION:...... 20
4.4.3 CLASSIFIER:...... 23
4.4.3.1 SUPPORT VECTOR MACHINE:...... 23
4.4.3.2 MAXIMUM ENTROPY:...... 24

5. EXPERIMENTAL RESULTS AND EVALUATION......25

5.1 EVALUATION METRICS:...... 25
5.2 RESULTS AND DISCUSSION:...... 27

6. CONCLUSION AND FUTURE WORKS......33

REFERENCES......43

LIST OF TABLES AND FIGURES

TABLE 1: SAMPLE BANGLA TWEETS 15
TABLE 2: EXPERIMENTAL RESULTS OF PRECISION, RECALL AND F-MEASURE FOR SUPPORT VECTOR MACHINE (SVM) BOLDFACE: BEST PERFORMANCE (IN F-MEASURE) FOR A CLASSIFICATION LABEL 29
TABLE 3: EXPERIMENTAL RESULTS OF PRECISION, RECALL AND F-MEASURE FOR MAXIMUM ENTROPY. BOLDFACE: BEST PERFORMANCE (IN F-MEASURE) FOR A CLASSIFICATION LABEL. ... 30
TABLE 4: COMPARISON OF THE ACCURACY RATES ON THE TEST DATA USING SVM AND MAXENT WITH VARIOUS SETS OF FEATURES. BOLDFACE: BEST PERFORMANCE (IN ACCURACY) FOR A GIVEN FEATURE (ROW)... 31
TABLE 5: ACCURACY OF SVM AND MAXIMUM ENTROPY IN [15] 32
TABLE 6: BANGLA POSITIVE POLARITY WORD LIST. NN, VM, JJ AND RB STAND FOR NOUN, VERB, ADJECTIVE AND ADVERB RESPECTIVELY. HIGHLIGHTED WORDS IN GRAY INDICATE TO THE INFLECTED AND MISSPELLED FORMS. 36
TABLE 7: BANGLA NEGATIVE POLARITY WORD LIST. NN, VM, JJ AND RB STAND FOR NOUN, VERB, ADJECTIVE AND ADVERB RESPECTIVELY. HIGHLIGHTED WORDS IN GRAY INDICATE TO THE INFLECTED AND MISSPELLED FORMS. 42

FIGURE 1: SYSTEM ARCHITECTURE 16
FIGURE 2: GENERAL BOOTSTRAPPING PROCESS USING LABELED AND UNLABELED DATA 20
FIGURE 3: ACCURACY RESULTS FOR SVM AND MAXENT USING DIFFERENT FEATURE SETS 32

ABSTRACT

Sentiment analysis has received great attention recently due to the huge amount of user-generated information on the microblogging sites, such as Twitter [1], which are utilized for many applications like product review mining and making future predictions of events such as predicting election results. Much of the research work on sentiment analysis has been applied to the English language, but construction of resources and tools for sentiment analysis in languages other than English is a growing need since the microblog posts are not just posted in English, but in other languages as well. Work on Bangla (or Bengali language) is necessary as it is one of the most spoken languages, ranked seventh in the world [13]. In this paper, we aim to automatically extract the sentiments or opinions conveyed by users from Bangla microblog posts and then identify the overall polarity of texts as either negative or positive. We use a semi-supervised bootstrapping approach for the development of the training corpus which avoids the need for labor intensive manual annotation. For classification, we use Support Vector Machines (SVM) and Maximum Entropy (MaxEnt) and do a comparative analysis on the performance of these two machine learning algorithms by experimenting with a combination of various sets of features. We also construct a Twitter-specific Bangla sentiment lexicon, which is utilized for the rule-based classifier and as a binary feature in the classifiers used. For our work, we choose Twitter as the microblogging site as it is one of the most popular microblogging platforms in the world.

1. Introduction

In recent years, microblogging sites have become a very popular source for publishing huge amount of user-generated information. One of the unique characteristics of these microblogging sites is that the messages that are posted by the users are short in length and users publish their views and opinions on different topics such as politics, religion, economics, business, and entertainment. These huge volumes of user-generated information on the microblogging sites are utilized for many applications. Product review mining is one such application where potential consumers go through the opinions expressed by previous consumers on different sites before acquiring a particular product or service, while companies analyze the feedbacks on different products or services posted by consumers on these sites to gain knowledge about which products or services to sell more and which should be improved. These microblogging sites are also used as a source of data for making future predictions of events, such as predicting election results. Here, we are not talking about going through just one or two user messages on a particular product or service and making a decision on that. Instead, millions of messages that are posted daily on the microblogging sites need to be checked, all the relevant posts for that product or service need to be extracted, different types of user opinions need to be analyzed, and finally the user opinions and feedbacks need to be summarized into useful information. For a human being, this is a very tedious and time-consuming work. This is where sentiment analysis comes in use.

Sentiment analysis or opinion mining is the automatic extraction of opinions, emotions, and sentiments from texts. Sentiments, opinions, and emotions are subjective impressions and not facts, which are objective or neutral. Through sentiment analysis, a given text can be classified into one of the three categories - positive, negative, or neutral. Sentiment analysis of texts can be performed at different levels like - document, sentence, phrase, word, or entity level. Since our domain is restricted to Microblogging sites, more specifically Twitter, as we only deal with Twitter corpus, we perform sentiment analysis at tweet level.

Much of the research work on polarity classification of Microblog posts has been implemented on the English language, but construction of resources and tools for sentiment analysis in languages other than English is a growing need since the microblog posts are not just posted in English, but in other natural languages as well. Work on other languages is growing, including Japanese ([2], [3], [4], [5]), Chinese ([6], [7]), German [8], and Romanian ([9], [10]). Much of the work on sentiment analysis for Bangla (or Bengali) language has been applied to the news corpus and blogs ([11],[12]), but we couldn't find any research paper which focuses on the issue of extracting user opinions and views from Bangla Microblog posts. As will be discussed in section 4.2 of our paper, characteristics of the microblog domain are quite different from blogs and news corpus, so there is an opportunity for research in this domain for the Bangla language.

In this paper, we aim to automatically extract the sentiment or polarity conveyed by users from Bangla microblog posts. We assume that the microblog posts are shared by users to express opinions and subjective content, therefore, restricting our classification problem to a binary classification problem of solely identifying the overall polarity of the text as either negative or positive. We use a semi-supervised bootstrapping approach for the development of the training corpus which avoids the need for labor intensive manual annotation. From related works in English, support vector machine (SVM) and Maximum Entropy (MaxEnt) have proven to outperform other classifiers in this field. Hence, for classification, we use SVM and MaxEnt and do a comparative analysis on the performance of these two machine learning algorithms by experimenting with a combination of various sets of features. We also construct a Twitter-specific Bangla sentiment lexicon. For our work, we choose Twitter as the microblogging site as it is one of the most popular microblogging platforms in the world

The rest of this paper is organized as follows: Section 2 describes the related work. Section 3 discusses Bangla sentiment lexicon construction. Section 4 presents proposed methodology and Section 5 explains the performance evaluation measures and focuses on the experimental results and discussion. Finally, results are summarized and concluded in section 6.

2. Related Works in English

We briefly overview the main lines of research carried out on the English language. There are a large number of approaches that has been developed to date for classifying sentiments or polarities in English texts. These methods can be classified into two categories- (1) machine learning or statistical-based approach and (2) unsupervised lexicon-based approach.

Machine learning methods use classifiers that learn from the training data to automatically annotate new unlabeled texts with their corresponding sentiment or polarity. [14] is one of first papers to apply supervised machine learning methods to sentiment classification. The authors perform the classification on movie reviews and show that MaxEnt and SVM outperform Naïve Bayes (NB) classifier. One of the first papers on the automatic classification of sentiments in Twitter messages, using machine learning techniques, is by [15].Through distant supervision, the authors use a training corpus of Twitter messages with positive and negative emoticons and train this corpus on three different machine learning techniques- SVM, Naïve Bayes, and MaxEnt, with features like N-grams (unigrams and bigrams) and Part of Speech (POS) tags. They obtain a good accuracy of above 80%. [16] follow the same procedures as [15] to develop the training corpus of Twitter messages, but they introduce a third class of objective tweets in their corpus and form a dataset of three classes- positive sentiments, negative sentiments, and a set of objective texts (no sentiments). They use multinomial NB, SVM, and Conditional Random Field (CRF) as classifiers with N-grams and POS-tags as features. The authors of [17] use 50 hashtags and 15 emoticons as sentiment labels to train a supervised sentiment classifier using the K Nearest Neighbors (KNN) algorithm. In [18], the authors implement a 2-step sentiment detection framework by first distinguishing subjective tweets from non-subjective tweets and then further classify the subjective tweets into positive and negative polarities. The authors find that using meta-features (POS tags) and tweet-syntax features (emoticons, punctuations, links, retweets, hashtags, and uppercases) to train the SVM classifiers enhances the sentiment classification accuracy by 2.2% compared to SVMs trained from unigrams only. Although supervised machine learning methods have been widely employed and proven effective in sentiment classification, they normally depend on a large amount of labeled data, which is both time consuming and labor intensive work.

Unsupervised lexicon-based methods rely on manually or semi-automatically constructed lexical resources, such as lexicons, to identify the overall polarity of texts. Lexicon is a collection of strong sentiment-bearing words or phrases, which are labeled with their prior polarity, or the context-independent polarity most commonly associated with the lexicon entries. There are several lexicons in English which are available online such as - ANEW [19], General Inquirer [20], OpinionFinder [21], SentiWordNet [22] and WordNet-Affect [23]. One of the initial works to apply unsupervised techniques to sentiment classification problem is by [24]. In the paper, a document is classified as positive or negative by the average semantic orientation of the phrases in the document that contain adjectives or adverbs. The semantic orientation of a phrase is calculated as the Pointwise Mutual Information (PMI) with a positive seed word “excellent” minus the PMI with a negative seed word “poor”. This approach achieves an accuracy of 84% for automobile reviews and 66% for movie review. In [25], the authors manually develop a sentiment lexicon consisting of positive and negative sentiment-bearing words annotated with their POS tags. This sentiment lexicon, along with a set of rules, is used to first classify the tweets as subjective or objective and then further classify the subjective tweets as positive, negative or neutral. They use a corpus of political tweets collected over the UK pre-election period in 2010. For the task of correctly identifying that a document contains a political sentiment and then correctly identifying its polarity, they get 62% Precision and predict 37% Recall. Other works addressing this lexicon-based approach include [26] and [27]. However, methods based on lexical resources often have the problem of obtaining low recall values because they depend on the presence of the words comprising the lexicon in the message to determine the orientation of opinion [44]. And due to the varied and changing nature of the language used on Twitter, this approach is not suitable for our thesis work. Moreover, as such lexical resources are not available for many other languages spoken in social media, like Bangla, hence this approach often becomes unsuitable for scarce-resource languages.

To overcome the problems of using a fully supervised machine-learning or unsupervised lexicon-based approach, some recent papers use a hybrid approach of employing both both lexicon and machine learning based approaches for their work. Works using hybrid approach include [53], [54] and [55].

For our thesis, we use a hybrid approach of firstly constructing a Bangla polarity lexicon that automatically labels a small training data, with the help of a rule-based classifier. We then identify the overall polarity of tweets using a semi-supervised machine learning approach.

3. Lexicon

As explained in section 2, several English polarity lexicons are available online, but we couldn't find any Bangla polarity lexicon to be used for our work. One approach is simply translating an existing English lexicon in Bangla language using a bilingual dictionary. Although this approach can easily be implemented, it does not create a high accuracy resource due to the highly overloaded meaning of words, as shown in [9] and [28]. So, we construct a Twitter-specific Bangla polarity lexicon, by leveraging on lexical resources already available for the English language, such as SentiWordNet.

3.1 Methodology to construct the Bangla sentiment lexicon:

In order to create a Bangla sentiment lexicon, which contains Bangla words annotated with their corresponding polarity (positive/negative) and Part-of-Speech (POS), we first construct an initial word list, containing strong positive and negative sentiment-bearing words, using a Twitter corpus with emoticons. The word list is then further expanded with the corresponding synonyms of the words in the wordlist.

To create the initial set of positive and negative word list, we collect two Bangla tweet corpora (these corpora were not used in training or test dataset) - one corpus contains tweets with all negative emoticons while the other corpus contains tweets with all positive emoticons. Emoticons are widely used by users in micro-blogging sites, like Twitter, to express emotion or sentiment about different topics. Hence, we assume that tweets with emoticons will hold strong positive and negative polarity words. All the tweets in the corpora are labeled as positive or negative according to their emoticon sign. We count the number of positive and negative emoticons for each tweet and annotate a tweet with the same polarity as the emoticon with the highest count. We use the emoticon list [29] for this task. All the words in the tweets are then marked with their POS tags and only adjectives, adverbs, nouns, and verbs are extracted, since previous studies on sentiment analysis ([30], [31]) have shown that these POS tags are strong indicators of subjectivity. All the extracted words are marked with positive or negative sentiment according to their tweet polarity. Note that the polarity given to the extracted words may not represent the actual sentiment of the words, since the polarity of the extracted words are merely assigned according to their tweet polarity (which is labeled using emoticons). Hence to verify whether the sentiment label that is given to each extracted word according to its tweet emoticon is actually its polarity, we further apply a filtering method using online bilingual dictionary and SentiWordNet. We firstly translate each extracted Bangla word in English using the online bilingual dictionaries Ovidhan [32] and Samsad [33] and then use SentiWordNet to check each of the translated word for its polarity, according to its POS. In SentiWordNet, a word with a specific POS can have many senses, and each sense is given 3 scores for positivity, negativity, and objectivity, indicating the strength of each of the three properties for that sense of the word. The 3 scores together are equal to 1. As we are dealing with positive and negative sentiment words, we only look at the positive and negative scores. So, if any sense for that POS of a word has a score greater than the threshold value of 0.6 for the corresponding polarity, then it is confirmed that the polarity of the word is actually that of the tweet in which it appeared and hence added in the initial word list. In this way the initial word list is generated. The reason for using a high threshold value of 0.6 is to include only the strong opinionated words in our lexicon. The word list is further expanded by adding the synonyms of the words using online available Bangla Wordnet of Indian Statistical Institute [34]. The synonyms are given the same polarity as the words they are expanded from. We finally add the inflected and misspelled forms of the words in the word list to make our lexicon Twitter-specific. In this way, a total of 737 single words made up the lexicon, which is provided in Tables 6 and 7.

We use this Bangla polarity lexicon for the rule-based classifier and feature extraction. This is explained in the next section in our approach.

4. Methodology

Our system architecture, outlining the whole process, is shown in figure 1 below.

4.1 DATASET:

Our dataset is a collection of Bangla tweets downloaded by querying Twitter REST API v1.1 [35] over a span from May-November 2013. As Twitter API supports language filtering and allows specifying the language of the retrieved posts, the optional language parameter in the Twitter Search URL was set to 'bn' to extract all Bangla tweets. Eventually, we collected a total of 1300 tweets by polling Twitter API. We split this dataset into training set and test set, comprising 1000 and 300 tweets respectively. However, the tweets we retrieved contained English text as well; instead of filtering out the English text from the tweets, we include them as part of our training and test sets. Because English words express strong positive and negative sentiment, they will likely contribute in sentiment classification of our dataset.

4.2 TWEET DESCRIPTION:

Twitter users communicate by publishing short messages called ‘tweets’. A tweet is limited to 140 characters; the language used in tweets is very informal and contains various misspellings, punctuations, slang, new words, URLs and genre-specific terminology and abbreviation, e.g. RT for retweets and #hashtag [50]. Examples of presence of such features in Bangla tweets are shown in Table 1. However, these unique features are not seen in blogs and news corpus where the posts are longer and more formal.

4.3 PREPROCESSING:

The raw tweets data obtained through Twitter API are noisy and hence, are preprocessed. We perform pre-processing through three steps – tokenization, normalization, and POS Tagging.

For tokenization, NLTK’s Tokenizer [36] package is used. Tweets contain special tokens such as username (e.g., @user), URL link, hashtag (e.g., #রাগ (#angry)) and retweets (e.g., RT). As these tokens do not express any sentiment, they are removed from the dataset. RT is a special token which denotes retweets; every occurrence of this token is removed from the dataset. Username and URL links are first replaced with ‘AT-USER’ and ‘URL’ tokens respectively; then the AT-USER and URL placeholders are removed. The hashtagged words are dealt with by removing the hashtag character and afterwards, treating the non-hashtagged word as a candidate token in the dataset that expresses some sentiment. As our dataset contains both Bangla and English tweets, punctuation marks for both are removed. Furthermore, all Bangla punctuations and their usage is similar to that of English (e.g.,’!’,’;’,’?’,’,’) ,except for dari (‘।’), which is bangla equivalent of full stop. Therefore, only dari is removed using Bangla Unicode in regular expression; for all other punctuation marks, English Unicode is used. Special care is taken to ensure that colons and semicolons are not removed as often Twitter users use emoticons (e.g., :), :(, ;)) to express sentiment; we use emoticons as a feature in feature vector during classification process and therefore need to keep these emoticons in our dataset. Similarly, all special characters (symbols not present in emoticons) are removed; parentheses are not eliminated to preserve the emoticons. Other normalization steps in pre-processing include: English tokens changed to lower case and English and Bangla stop words removed as these only serve functional purpose, but express no sentiment. Elongated words (e.g., মহাননন (greattt)) are also identified and corrected using relevant regular expression; this is done for both Bangla and English words with character repetitions.

POS Tagging completes the preprocessing of tweets. POS Tagging of English tokens is done using NLTK POS-Tagger [37]. For Bangla, POS Tagging is performed by the Bangla Pos-Tagger Package [38]. An advantage of this pos-tagger is that it implements NLTK'S Brill Tagger; Brill Tagger works by first being trained on an initial POS-Tagger, and then based on transformation rules improves the tagging. As our corpus is Twitter specific, we provided our manually pos-tagged tweets, labeled using Bengali Shallow Parser [43], as the initial pos-tagger. As a result, Bangla tokens are pos-tagged more accurately.

কি চমৎকার দেখা গেল... RT: @gf: হাসতেই আছি হাসতেই আছি :)) xXxD http://kcy.me/3ud8 #GoogleTranslate #fb
@Kd হুম Samsung netbook এর performance তুলামুলকভাবে ভালই মনে হচ্ছে। @me @dee
৬০,০০০ টাকার মোবাইল ফোনে Bluetooth নাই। তাইলে সেই আপেল হাতে না রেখে খাওনটা ভালা। #iPhone #Apple
♥ ♥ ♥ HP Pavilion DV7-7003TX ♥ ♥ ♥ বাজেট ছাড়া সবকিছুই ভাল লাগসে !!!
আমার নেটবুক থেকে মাঝেমাঝে মৃদু গুড়গুড় আওয়াজ আসতেছে। ঘটনা কি? :S
@ssmm :P http://bit.ly/nRVYUX মোবাইলের থিম রিলিজ দিসে, তাও এই! এন্ড্রয়েড ই ভালা :D
রাজধানির মিরপুরের কাজীপারায় laptop বিস্ফরিত হয়ে এক জুবক মারা গেছেন। :(http://prothom-alo.com......
শুদ্ধ রূপে বাংলা শিখিবার জন্য #googletranslate এর বিকল্প নাই :p:P

Table 1: Sample Bangla Tweets

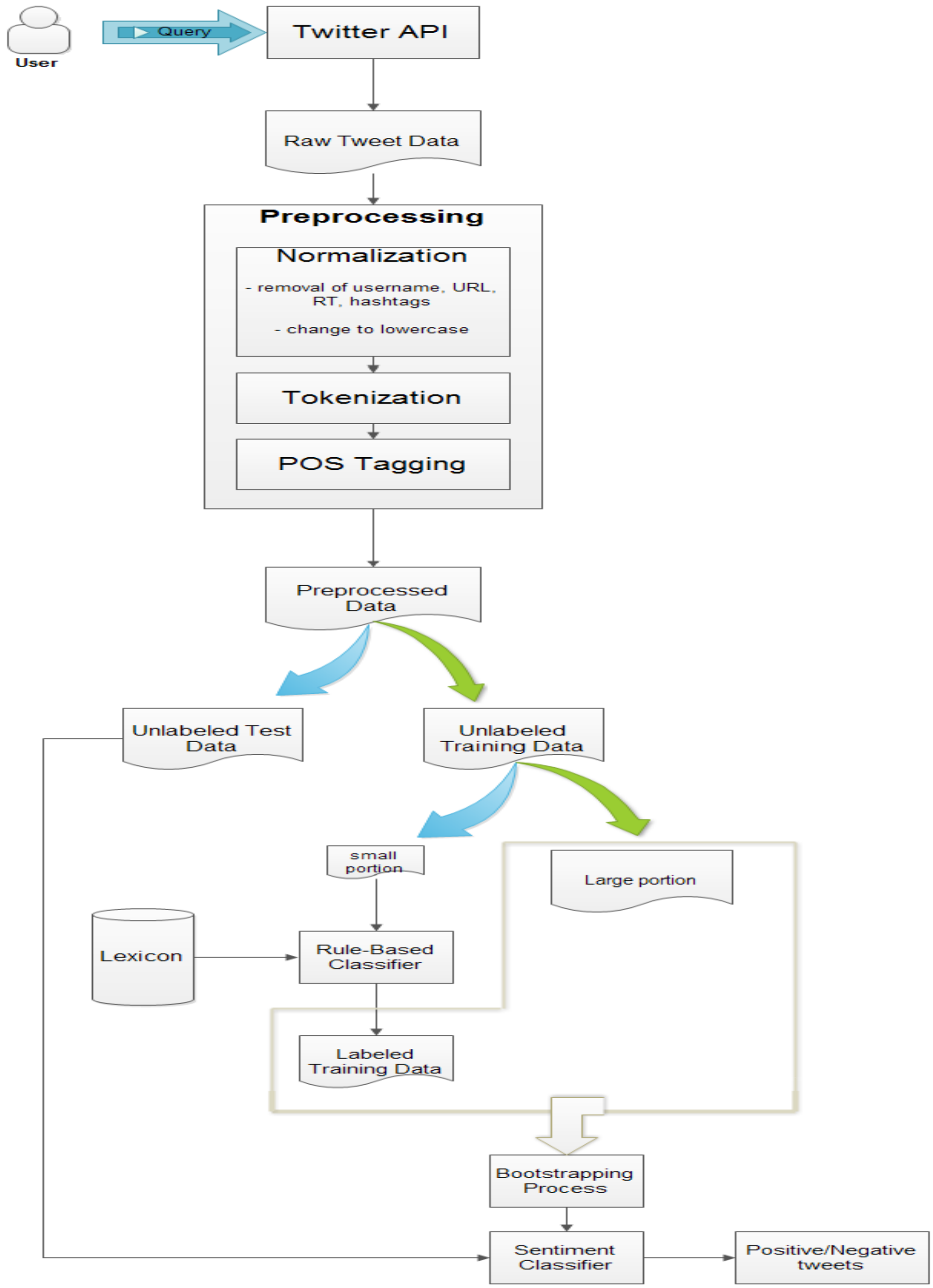

Figure 1: System Architecture

4.4 Our Approach

4.4.1 TRAINING SET CONSTRUCTION:

4.4.1.1 SEMI-SUPERVISED:

Semi-supervised refers to the machine learning techniques that include the unlabeled data along with the labeled data in the training process. As large amount of labeled data is often difficult to obtain while unlabeled data are in abundant, therefore semi-supervised learning techniques address this issue by incorporating unlabeled data in the training process to further improve the classification performance [39].

Traditional classifiers learn from large labeled data during training – this is known as supervised learning. Due to the lack of large labeled Bangla tweet dataset, the supervised learning approach is not suitable for this problem. Although it is possible to manually label the Bangla tweets, it becomes very labor-intensive and time consuming.

Whereas, although unsupervised methods work with unlabeled data, but the overhead is the need for a large domain-dependent Bangla sentiment lexicon or dictionary, which is not readily available for scarce-resource languages like Bangla.

In our case, we have a somewhat large unlabeled bangla tweet dataset available. Hence our choice is a semi-supervised approach as it builds better classifiers using large unlabeled data along with a small set of labeled data under appropriate assumptions ([39], [40]).

Among the different semi-supervised learning methods, we use self-training, which is a commonly used technique for semi-supervised learning.

4.4.1.2 SELF-TRAINING BOOTSTRAPPING:

Self-training bootstrapping is performed to develop our labeled training data set. Self-training bootstrapping works by first labeling a small dataset, then a classifier is trained on that small labeled data, and afterwards the trained classifier is applied on the set of unlabeled data. Instances of the unlabeled data, together, with their predicted labels, are added to the labeled training data [39]. The classifier is then retrained on this newly labeled data, and the process is repeated for several iterations.

So, following the steps of self training bootstrapping, at first we utilize the manually constructed Bangla lexicon, containing strong positive and negative sentiment bearing Bangla words, to label a small number of Bangla tweets based on a certain rule. Since our collected Bangla tweets also contain some English words in them, we use the online available English lexicon [56], which consists of around 6800 positive and negative sentiment words. We decided to use this specific English lexicon [56], because unlike other existing English lexicons, this lexicon consists of many misspelled words which appear frequently in social media content like tweets.

To make this rule-based classifier, we set the following rules, keeping in mind that tweets are short (restricted to 140 characters):

1. If $count_{positive} > count_{negative}$:

   Label 'positive'

2. If $count_{negative} > count_{positive}$:

   Label 'negative'

That is, words in tweets are compared with the words in the Bangla and Enlgish lexicons for a match. For every match, it is then checked if the word has a positive or a negative label in the polarity lexicons, and count for that label is incremented accordingly. If the count of positive words exceeds that of negative words, the Bangla tweet is labeled as positive and vice versa. Tweets without any lexicon entry or with equal count of positive and negative words are discarded for our experiment.

Applying these rules, we got a small labeled dataset of 100 tweets maintaining an equal class ratio of 50 positive and 50 negative.

So this way, our initial small set of unlabeled tweets got labeled to be used to develop the training set.

Next, we train the classifier on this small labeled set of tweets. Once the classifier is trained and learns the patterns, it is applied to the unlabeled dataset. Our total unlabeled dataset has a size of 1000 tweets. For the first iteration of bootstrapping, the trained classifier is applied on 50 unlabeled tweets.

However, only those tweets with a high confidence are added to the labeled training set and the rest are cut off and added back to the unlabeled data set. To determine the tweets with confidence above a certain threshold, we find probability distribution over labels for the given feature set for each tweet and pick those 30 tweets out of the 50 labeled tweets as highly confident which have the highest probability distribution. We use probability distribution as it tells how likely the classifier thinks that a tweet belongs to a particular class/label.

Now this newly labeled bigger dataset is used to retrain the classifier, which is then applied on another portion of the unlabeled dataset. This way, we repeat the self-training bootstrapping until all the 1000 unlabeled tweets are labeled.

One important point for the bootstrapping process is that for each iteration, class distribution in the labeled data is maintained by keeping a constant ratio across classes between already labeled examples and newly added examples [41].

Figure 1 illustrates the general bootstrapping process, taken from [41].

0. Given:
- A set L of labeled training examples
- A set U of unlabeled examples
- Classifiers C_i

1. Create a pool U' of examples by choosing P random examples from U
2. Loop for I iterations:
    2.1 Use L to individually train the classifiers C_i, and label the examples in U'
    2.2 For each classifier C_i select G most confidently examples and add them to L, while maintaining the class distribution in L
    2.3 Refill U' with examples from U, to keep U' at a constant size of P examples

Figure 2: General bootstrapping process using labeled and unlabeled data

4.4.2 FEATURE EXTRACTION:

In feature extraction, each tweet is represented as a set of features called a feature vector. Feature extraction is done on the training set developed, in order to use the extracted features in the training process to train the sentiment classifier.

The following set of features was used for each tweet:

<u>Word N-Gram</u>

Each tweet is represented as a contiguous sequence of N tokens called an N-gram. We use unigrams and bigrams for our work.

In unigram, N=1 and the presence/absence of each individual Bangla token in the tweet is the feature. For example, for the Bangla tweet

আমার ভাল লাগতাছে না

(English translation) I am not feeling well

The unigram representation is [‘আমার’, ‘ভাল’, ‘লাগতাছে’, ‘না’].

Similarly, in bigram, N=2 and every sequence of two adjacent words for each tweet is extracted as the feature.

For the Bangla tweet above the bigrams are,

[[‘আমার’, ‘ভাল’], [‘ভাল’, ‘লাগতাছে’], [‘লাগতাছে’, ‘না’]]

Emoticon

Emoticons are the use of letters and symbols to convey facial expressions. In previous works [16], emoticons have shown to express strong positive or negative sentiment and thereby, proved very useful to sentiment analysis.

The Bangla tweets we retrieved also contain emoticons; so existence of emoticon token in the tweet is used as a feature.

The regular expression used to extract emoticons from tweets during preprocessing is adopted from Christopher Potts’ tokenizing script [42].

We use the emoticon polarity dictionary developed by Leebecker et al [29] as our emoticon lexicon. In that emoticon polarity dictionary, each emoticon is given a sentiment score ranging from -2 to 2. Instead of using sentiment score, we annotate the emoticons with their respective positive or negative sentiment label according to their sentiment score. Emoticons with positive sentiment score are mapped to positive; similarly, emoticons with negative sentiment score are mapped to negative.

Lexicon

As our Bangla polarity lexicon and the English lexicon [56] contain strong positive and negative sentiment expressing words, we use the word entries in the lexicons as features.

The presence/absence of each entry in the lexicons is checked per tweet.

Pos-Tagging

Each individual token in the tweet is appended with a part of speech tag using the POS Tagger discussed in section 4.3. We use POS tagging along with lexicon as a combined feature. This feature is implemented in the same way as the lexicon feature, but instead of just matching each lexicon word entry, both the lexicon word and its part of speech tag need to match with the POS tagged tokens of tweet.

Negation

Although negation words don't express any sentiment, they affect the overall sentiment of a tweet. Use of negation in Bangla is different from that in English. Unlike English, where negative words usually occur in the middle of a sentence, Bangla sentences frequently contain negation toward the end. An example is the tweet

আমার ভাল লাগতাছে না

(English translation) I am not feeling well

Hence, we didn't follow the negation handling method specified in [14], where every word following the negation word is appended with a 'NEG' suffix to reverse its sentiment.

Instead, we manually construct our own negation word list and use it as a binary feature.

Our Bangla negation list is the following,

["নাই", "না", "নয়", "নাআ", "নেই" , "নহে" , "নায়" , "নেয়"]

As Bangla is a highly inflected language, we include all the inflected forms of the base negation words as well.

Combinations of Features

We use the above mentioned features in combinations, with the expectation of better results. The following combinations were used:

- Unigram + Bigram
- Unigram + Negation
- Unigram + Emoticon
- Unigram + Negation + Emoticon
- Unigram + Lexicon
- Unigram + Lexicon + POS
- All

4.4.3 CLASSIFIER:

After the training process, we then apply the trained classifier on the test dataset of 300 tweets so that new tweets are labeled as positive or negative.

In our work, we use two state-of-the-art classifiers, namely, Support Vector Machine and Maximum Entropy. Both have shown to be effective in previous text categorization studies [14].

4.4.3.1 SUPPORT VECTOR MACHINE:

Support Vector Machine (SVM) classifiers are non-probabilistic binary linear classifiers, that is, for a given set of training tweets, where each tweet is marked as belonging to the positive sentiment class or the negative sentiment class, an SVM training algorithm builds a model that assigns new test tweets into one of the two possible classes. The basic idea of the classifier is to find a hyper-plane that separates the positive and negative sentiment classes with maximum margin (or the largest possible distance from both classes).

We use LIBSVM [45] package in python with a linear kernel for training and testing the Bangla Twitter data, with all parameters set to their default values.

4.4.3.2 MAXIMUM ENTROPY:

Maximum Entropy (MaxEnt or ME) is a probabilistic classifier which falls under the category of exponential models. Maximum Entropy has occasionally shown to give better results than Naïve Bayes classifier for text classification [46]. This is because, unlike Naïve Bayes, Maximum Entropy does not make assumption that features are conditionally independent. This is true for sentiment classification of tweets, where the word features comprising of individual word tokens obviously are not independent of each other. Hence we conducted our experiments using Maximum Entropy classifier.

Maximum Entropy model finds weights for the features to maximize the log-likelihood of the training data in order to improve the model. The probability of class *c* given a tweet *d* and weights λ is given by the following formula:

$$P(c|d,\lambda) \stackrel{def}{=} \frac{\exp \sum_i \lambda_i f_i(c,d)}{\sum_{c' \in C} \exp \sum_i \lambda_i f_i(c',d)}$$

Our experiments are performed using MaxentClassifier [47] in NLTK's Classify package. We use maximum iteration as the cut off parameter, setting it to five. Although more iterations generally improve accuracy [48], but due to the small size of our dataset, further iterations did not have any additional effect on the accuracy obtained at five iterations.

5. Experimental Results and Evaluation

5.1 EVALUATION METRICS:

In order to evaluate the performance of the two learning algorithms used, namely Maximum Entropy and SVM, we first use the standard precision, recall and F-measure to measure the positive and negative sentiments for each classifier using various sets of features. We then use the accuracy metric to compare the overall performance of the two classifiers. Sentiment analysis task can be interpreted as a classification task where each classification label represents a sentiment. Hence, we define and calculate the four metrics for each label (positive and negative) the same way as in general classification task.

In a classification task, precision, recall, F-measure and accuracy are explained using four terms - true positive, true negative, false positive and false negative.

True Positive (tp) is defined as the number of tweets, from the test set, correctly labeled by the classifier as belonging to a particular class or label.

True Negative (tn) is defined as the number of tweets, from the test set, correctly labeled by the classifier as not belonging to a particular class or label.

False Positive (fp) is defined as the number of tweets, from the test set, incorrectly labeled by the classifier as belonging to a particular class or label.

False Negative (fn) is defined as the number of tweets, from the test set, that are not labeled by the classifier as belonging to a particular class or label but should have been.

We now define the evaluation metrics using these four terms as follows:

Precision is the number of tweets in the test set that is correctly labeled by the classifier from the total tweets in the test set that are classified by the classifier for a particular class. That is,

$$\text{Precision (P)} = \frac{tp}{tp+fp}$$

Recall is the number of tweets in the test set that is correctly labeled by the classifier from the total tweets in the test set that are actually labeled for a particular class. That is,

$$\text{Recall (R)} = \frac{tp}{tp+fn}$$

F-measure is the weighted harmonic mean of precision and recall for a particular class. That is,

$$\text{F-measure} = \frac{2*P*R}{P+R}$$

Accuracy is the percentage of tweets in the test set that the classifier correctly labels. That is,

$$\text{Accuracy (A)} = \frac{tp+tn}{tp+tn+fp+fn} * 100\%$$

To calculate precision, recall, F-Measure and accuracy, we use NLTK metrics module [49], which provides functions for calculating these metrics. For each classification label, we passed two sets, a reference set and a test set, as arguments into the NLTK metrics module functions- precision(), recall(), f_measure() and accuracy(). Reference set contains all the correctly labeled tweets while the test set contains the tweets classified by the classifier for a particular label.

5.2 RESULTS AND DISCUSSION:

We provide the experimental results of precision, recall, and F-measure for the binary classification task using SVM and Maximum Entropy with various sets of features in Table 2 and Table 3. In Table 2, we can see that for SVM, we get the best F-measure score of 0.93, for both the positive as well as the negative sentiment label, when using a combination of unigrams and emoticons as feature. We can also see from Table 2 that using unigrams only or with other features like negation, bigrams, or lexicons give a score between 0.65-0.71, but when emoticons are added, there is an increase in F-measure by about 36.76% for both the sentiment labels. From this we can deduce that emoticon features play a crucial role in automatically identifying the sentiments of tweets, as Twitter users tend to express their feelings and opinions more with these emoticons than with words (n-grams) due the very short length of the tweets. We can see this same phenomenon in the case of Maximum Entropy in Table 3, which gives best score of 0.83 for positive sentiment label and 0.85 for negative sentiment label on the inclusion of emoticons with other features.

The accuracies of SVM and Maximum Entropy for the various feature sets used are given in Table 4, and we compare the overall performance, in accuracy, of the two classifiers in Figure 2. We can see from the table that the results of SVM and Maximum Entropy for all the sets of features are comparable, and overall, SVM slightly outperforms Maximum Entropy. Although for some features, Maximum Entropy gives slightly higher accuracy than SVM. We achieve the best accuracy of 93% for SVM when using unigrams with emoticons as feature. As explained above for Table 2 and Table 3, inclusion of emoticons feature greatly increases the performance of the classifiers as Twitter users tend to frequently use these emoticons to express subjective content in the short tweets. From the table, we observe that the lexicon features perform averagely for both the classifiers.

Inclusion of English lexicon features in addition to the Bangla lexicon features increase the accuracy of SVM by 7% and Maximum Entropy by 8%. Moreover, when unigram features are included with English and Bangla lexicon features, there is an additional increase in the accuracy by 5% for SVM and remain same for Maximum Entropy. But we observe a decrease in accuracy for both the classifiers when adding POS-based features with unigrams and lexicon features. This observation is consistent with several works in English Twitter sentiment analysis like [51], [52] and [15].

Since we couldn't find any published baseline results for sentiment analysis on Twitter data for the Bangla language, we use related works on the English language to evaluate the performance of our classifiers. We compare our experimental results with [15], which is one of the first papers on this field on the English Twitter data. Table 5 shows the experimental results of [15] and we see that the authors only use 4 features- unigram, bigrams, unigrams with bigrams, and parts of speech to train SVM and Maximum Entropy. As we use many more features than given in Table 5, so we decided to evaluate our results only on the basis of the four features used by [15]. We can see that for all four features, our results given in Table 4 are lower than those reported in Table 5 by [15]. One reason for this could be that [15] use a much larger training dataset of 1,600,000 tweets compared to our one, which only comprised of 1000 training tweets, and as the unigram, bigram and POS features solely depend on the presence of words in tweets, hence our smaller training data gives lower accuracy rates for these four features. We examine that like [15], we achieve worst performance using bigrams for both SVM and Maximum Entropy. As stated in [15], using only bigrams as features is not useful because the feature space is very sparse. So, the authors suggested that it is better to combine unigrams and bigrams as features. In Table 4, we observe an improvement in the accuracy when using both unigrams and bigrams as features. For SVM, we see an increase in accuracy by 9% and for Maximum Entropy, there is a gain of 18% over the bigram feature accuracy. From Table 5, we can see that there is a drop in accuracy when using unigrams and POS as features, which is the same in our case as well, as shown in Table 4. Although [15] didn't use emoticons as features in their binary classification task, they suggested in their future work that emoticon features are very valuable and it would be useful to take emoticons into account when classifying test data. From our experimental results in Table 4, we can see that the use of emoticon features is indeed very useful and gives promising performance for both the classifiers.

Support Vector Machine (SVM)										
No.	Features	Positive Sentiment			Negative Sentiment			Average		
		Precision	Recall	F-Measure	Precision	Recall	F-Measure	Precision	Recall	F-Measure
1	Unigram	0.68	0.63	0.65	0.66	0.70	0.68	0.67	0.67	0.67
2	Bigram	0.56	0.91	0.69	0.76	0.29	0.42	0.66	0.60	0.56
3	Unigram + Bigram	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69
4	Unigram + Negation	0.68	0.64	0.66	0.66	0.70	0.68	0.67	0.67	0.67
5	Emoticon	0.81	1.0	0.89	1.0	0.77	0.87	0.91	0.89	0.88
6	Unigram +Emoticon	0.92	0.94	**0.93**	0.94	0.91	**0.93**	0.93	0.93	0.93
7	Unigram + Negation +Emoticon	0.91	0.94	0.92	0.94	0.91	0.92	0.93	0.93	0.92
8	Lexicon (Bangla)	0.55	0.97	0.71	0.89	0.21	0.34	0.72	0.59	0.53
9	Lexicon (English + Bangla)	0.85	0.38	0.53	0.60	0.93	0.73	0.73	0.66	0.63
10	Unigram + Lexicon (English + Bangla)	0.72	0.71	0.71	0.71	0.72	0.71	0.72	0.72	0.71
11	Unigram + Lexicon (English + Bangla + POS)	0.58	0.93	0.71	0.83	0.32	0.47	0.71	0.63	0.59
12	All	0.79	1.0	0.89	1.0	0.74	0.85	0.89	0.87	0.87

Table 2: Experimental results of Precision, Recall and F-measure for Support Vector Machine (SVM) Boldface: best performance (in F-measure) for a classification label

Maximum Entropy (MaxEnt)										
No.	Features	Positive Sentiment			Negative Sentiment			Average		
		Precision	Recall	F-Measure	Precision	Recall	F-Measure	Precision	Recall	F-Measure
1	Unigram	0.69	0.65	0.67	0.67	0.72	0.69	0.68	0.69	0.68
2	Bigram	1.0	0.017	0.034	0.50	1.0	0.67	0.75	0.51	0.35
3	Unigram + Bigram	0.71	0.64	0.67	0.67	0.74	0.70	0.69	0.69	0.69
4	Unigram + Negation	0.69	0.65	0.67	0.67	0.72	0.69	0.68	0.69	0.68
5	Emoticon	0.99	0.59	0.74	0.71	0.99	0.83	0.85	0.79	0.79
6	Unigram +Emoticon	0.88	0.79	**0.83**	0.81	0.89	**0.85**	0.85	0.84	0.84
7	Unigram + Negation +Emoticon	0.88	0.79	**0.83**	0.81	0.89	**0.85**	0.85	0.84	0.84
8	Lexicon (Bangla)	0.56	0.96	0.71	0.85	0.25	0.38	0.71	0.60	0.55
9	Lexicon (English + Bangla)	0.85	0.43	0.57	0.62	0.92	0.74	0.74	0.68	0.66
10	Unigram + Lexicon (English + Bangla)	0.69	0.65	0.67	0.67	0.72	0.69	0.68	0.69	0.68
11	Unigram + Lexicon (English + Bangla + POS)	0.58	0.92	0.71	0.81	0.32	0.46	0.70	0.62	0.59
12	All	0.88	0.79	**0.83**	0.81	0.89	**0.85**	0.85	0.84	0.84

Table 3: Experimental results of Precision, Recall and F-measure for Maximum Entropy. Boldface: best performance (in F-measure) for a classification label.

No.	Features	Support Vector Machine (SVM)	Maximum Entropy (MaxEnt)
		Accuracy Rate %	Accuracy Rate %
1	Unigram	67	**68**
2	Bigram	**60**	51
3	Unigram + Bigram	**69**	**69**
4	Unigram + Negation	67	**68**
5	Emoticon	**88**	79
6	Unigram +Emoticon	**93**	84
7	Unigram + Negation +Emoticon	**92**	84
8	Lexicon (Bangla)	59	**60**
9	Lexicon (English + Bangla)	66	**68**
10	Unigram + Lexicon (English + Bangla)	**71**	68
11	Unigram + Lexicon (English + Bangla + POS)	**63**	62
12	All	**87**	84

Table 4: Comparison of the accuracy rates on the test data using SVM and MaxEnt with various sets of features. Boldface: best performance (in accuracy) for a given feature (row).

features	SVM	Maximum Entropy
Unigram	82.2	80.5
Bigram	78.8	79.1
Unigram + Bigram	81.6	83.0
Unigram + POS	81.9	79.9

Table 5: Accuracy of SVM and Maximum Entropy in [15]

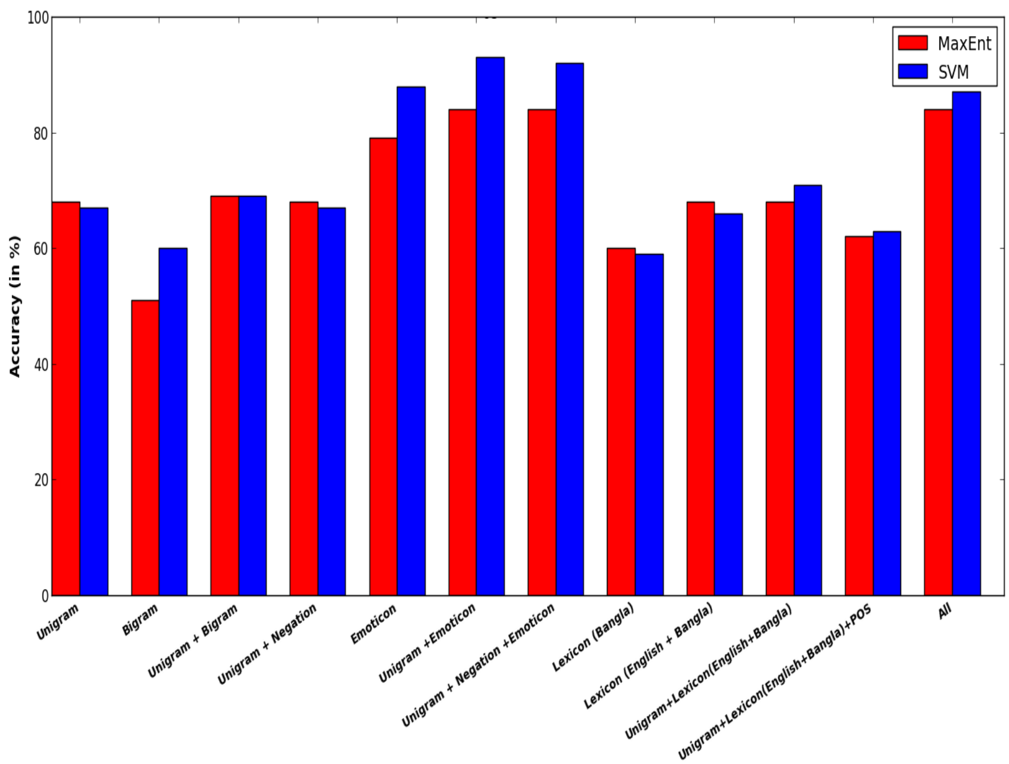

Figure 3: Accuracy Results for SVM and MaxEnt using different feature sets

6. Conclusion and Future Works

In this paper, we discuss how we generate the training data in a bootstrapping semi-supervised way and perform sentiment analysis on Bangla Twitter data. We present our empirical results on the test data using two machine learning classifiers with various sets of features. The results of our experiments on the whole are encouraging for the task of detecting sentiment in Bangla microblog posts. We achieve a satisfying accuracy of 93% for SVM using unigrams and emoticons as features. We observe from our results that emoticons used as features play a crucial role in training the classifiers for the binary classification task. N-gram features (unigrams and bigrams) perform poorly in comparison to the baseline results in [15]. One of the reasons for this could be due to the smaller training dataset used for our experiments. Hence in future work, we plan to improve the classification accuracy by using a larger training dataset. For our work, we assumed that tweets are subjective and hence didn't deal with the neutral class. But in real world, there exists objective tweets which do not express any sentiment and fall into the neutral class. For example, the following tweet, *যাত্রীর ল্যাপটপ থানায় জমা দিল ট্যাক্সি চালক* (*the taxi driver handed over the passenger's laptop to the police station*), neither expresses a positive nor a negative sentiment, and hence it is classified as a neutral tweet. So, we need to handle these neutral tweets in future.

		POSITIVE		Total 280 words			
Noun		Verb		Adjective		Adverb	
অনাবিলতা	NN	আকর্ষণ	VM	অকুণ্ঠ	JJ	অন্তরঙ্গভাবে	RB
অনুতোষ	NN	আকৃষ্ট	VM	অন্তরঙ্গ	JJ	অনুকূলভাবে	RB
অনুরাগ	NN	আশা	VM	অনুপ্রেরণীয়	JJ	ঘনিষ্ঠভাবে	RB
অভিনন্দন	NN	উপভোগ	VM	অপরূপ	JJ	নিপুণভাবে	RB
আকর্ষণ	NN	কামনা	VM	অপূর্ব	JJ	প্রিভাবে	RB
আত্মাদর	NN	পছন্দ	VM	অভিজ্ঞ	JJ	প্রীতিভরে	RB
আত্মমর্যাদা	NN	প্রত্যাশা	VM	অসাধারণ	JJ	ভালোভাবে	RB
আত্মসম্মান	NN	প্রশংসা	VM	অসাধারন	JJ	ভালভাবে	RB
আত্মসম্মানবোধ	NN	বিস্ময়বিমুগ্ধ	VM	আকর্ষণীয়	JJ	মর্যাদাসম্পন্ন	RB
আদর	NN	মজা	VM	আদর্শ	JJ	সঙ্গতভাবে	RB
আদরে	NN	শ্রদ্ধা	VM	আন্তরিক	JJ	সুবিধাজনকভাবে	RB
আদর্শ	NN	শ্রদ্ধাবোধ	VM	আনন্দকর	JJ		
আন্তরিকতা	NN	স্নেহ	VM	আনন্দিত	JJ		
আনন্দ	NN	সন্তোষজনক	VM	আনন্দজনক	JJ		
আনন্দগুলো	NN	ভালোবাসা	VM	আনন্দদায়ক	JJ		
আনন্দে	NN	ভালোবাসি	VM	আনন্দপূর্ণ	JJ		
আনন্দের	NN			আনন্দময়	JJ		
আনন্দলাভ	NN			আনন্দময়ী	JJ		
আবেগ	NN			আশাবাদী	JJ		
আমোদ	NN			উচিত	JJ		
আহ্লাদ	NN			উত্তম	JJ		
উপকার	NN			উৎসাহী	JJ		
উল্লাস	NN			উৎসাহনীয়	JJ		
উল্লাসে	NN			উদার	JJ		
উল্লাসের	NN			উপকারী	JJ		
করুণা	NN			উপযুক্ত	JJ		
কল্যাণ	NN			ঐশ্বরিক	JJ		
কল্যাণে	NN			কল্পনাপূর্ণ	JJ		
কুশলতা	NN			কল্পনাপ্রসূত	JJ		
ক্ষমতা	NN			কুশলী	JJ		
খুশি	NN			খুশি	JJ		
খুশিতে	NN			খুশির	JJ		
খুশির	NN			খোশ	JJ		
খোশ	NN			গৌরবান্বিত	JJ		
খুশীতে	NN			গৌরবজনক	JJ		
যোগ্যতা	NN			গুরুত্বপূর্ণ	JJ		
তারিফ	NN			গুরতপুনু	JJ		
তৃপ্তি	NN			ঘনিষ্ঠ	JJ		
তৃপ্তির	NN			চমৎকার	JJ		
তুষ্টি	NN			জয়ী	JJ		
তুষ্টিবিধান	NN			জয়যুক্ত	JJ		
তুষ্টিসাধন	NN			জব্বর	JJ		
দক্ষতা	NN			যোগ্য	JJ		
ধৈর্য্য	NN			ঠিক	JJ		
নির্মলতা	NN			দক্ষ	JJ		

ন্যায়পরায়ণতা	NN			দারুন	JJ		
পিরিত	NN			দ্বিধাহীন	JJ		
পছন্দ	NN			দুর্দান্ত	JJ		
পবিত্রতা	NN			ধন্য	JJ		
পরিতোষ	NN			নিঃস্বার্থ	JJ		
পরিতৃপ্তি	NN			নিপুণ	JJ		
পরিহার	NN			নিরোগ	JJ		
প্রীতি	NN			ন্যায্য	JJ		
প্রেমান্ধতা	NN			ন্যায়পরায়ণ	JJ		
প্রেম	NN			ন্যায়পর	JJ		
প্রণয়	NN			পটু	JJ		
পরমানন্দ	NN			পূণ্য	JJ		
প্রশংসা	NN			পবিত্র	JJ		
প্রশংসন	NN			পবিত্রীকৃত	JJ		
প্রশান্তি	NN			প্রিয়	JJ		
প্রশস্তি	NN			প্রীতিকর	JJ		
ফূর্তি	NN			প্রীতিপূর্ণ	JJ		
ফূর্তির	NN			পরম	JJ		
ভালো	NN			পরমোৎকৃষ্ট	JJ		
ভাল	NN			প্রশংসনীয়	JJ		
ভালু	NN			বিজয়ী	JJ		
ভালই	NN			বিমুগ্ধ	JJ		
ভালা	NN			বিস্ময়কর	JJ		
ভালাই	NN			বহুমূল্য	JJ		
ভালোই	NN			ভাগ্যবান	JJ		
ভালোর	NN			ভালো	JJ		
ভালোয়	NN			ভাল	JJ		
ভালোবাসা	NN			ভালু	JJ		
ভালোবাসার	NN			ভালই	JJ		
ভালোবাসায়	NN			ভালা	JJ		
ভালবাসা	NN			ভালাই	JJ		
ভালুবাসা	NN			ভালোই	JJ		
ভালবাসার	NN			ভালোয়	JJ		
ভালবাসায়	NN			মঙ্গলকর	JJ		
ভরসা	NN			মঙ্গলজনক	JJ		
মঙ্গল	NN			মঙ্গলদায়ক	JJ		
মায়া	NN			মায়াবী	JJ		
মজা	NN			মধুময়	JJ		
মজাই	NN			মধুর	JJ		
মজাও	NN			মনোরম	JJ		
মজার	NN			মনোমুগ্ধকর	JJ		
মজায়	NN			মনোহর	JJ		
মমতা	NN			মহান	JJ		
মমত্ব	NN			মহিমময়	JJ		
মর্যাদা	NN			মহীয়ান	JJ		
মস্তি	NN			লক্ষী	JJ		
মস্তির	NN			লক্ষ্মী	JJ		

শুভ	NN			লাভজনক	JJ		
শ্রদ্ধা	NN			শান্তির	JJ		
সক্ষমতা	NN			শোভন	JJ		
সুখ	NN			শুভ	JJ		
সুখই	NN			শ্রেষ্ঠ	JJ		
সুখিই	NN			সক্ষম	JJ		
সুখীই	NN			সুখী	JJ		
সুখে	NN			সুখি	JJ		
সুখের	NN			সুখিত	JJ		
সাধুতা	NN			সাদাসিধে	JJ		
সাধুতার	NN			সাধু	JJ		
সামর্থ্য	NN			সাধুসুলভ	JJ		
সোহাগ	NN			সাবাশ	JJ		
সত্যবাদিতা	NN			সৌভাগ্যশালী	JJ		
সততা	NN			সেরা	JJ		
সততার	NN			সঠিক	JJ		
সৎস্বভাব	NN			সৎ	JJ		
স্নেহ	NN			সতী	JJ		
স্নেহার্দ্রতা	NN			সত্যবাদী	JJ		
সন্তোষ	NN			সুদক্ষ	JJ		
সন্তুষ্টি	NN			সুদৃশ্য	JJ		
স্ফুর্তি	NN			সন্তুষ্ট	JJ		
স্ফুর্তির	NN			সুন্দর	JJ		
স্বাচ্ছল্য	NN			সুনদর	JJ		
সুবিধা	NN			সুন্দরি	JJ		
সম্মান	NN			স্পষ্ট	JJ		
হিত	NN			সফল	JJ		
জোশ	NN			স্বার্থশূন্য	JJ		
				সুবিচারপূর্ণ	JJ		
				স্বর্গসুখপ্রাপ্ত	JJ		
				সম্মানিত	JJ		
				সম্মানীয়	JJ		
				সম্মান্য	JJ		
				সমর্থ	JJ		
				সর্বশ্রেষ্ঠ	JJ		
				সরল	JJ		
				সুস্থ	JJ		
				সসুন্দর	JJ		
				সুস্বাদু	JJ		
				সহজ	JJ		
				সহজেই	JJ		

Table 6: Bangla positive polarity word list. NN, VM, JJ and RB stand for Noun, Verb, Adjective and Adverb respectively. Highlighted words in gray indicate to the inflected and misspelled forms.

		NEGATIVE		Total 457 words			
Noun		Verb		Adjective		Adverb	
অত্যাহিত	NN	অন্যায়	VM	অকারণ	JJ	দুঃখজনকভাবে	RB
অতৃপ্তি	NN	অপকার	VM	অকল্যাণসূচক	JJ	দুঃখদায়কভাবে	RB
অনাচার	NN	অপছন্দ	VM	অক্ষম	JJ	দুরদৃষ্টক্রমে	RB
অনিষ্ট	NN	অপমান	VM	অচেনা	JJ	দুর্ভাগ্যক্রমে	RB
অন্যায়াচরণ	NN	অপমানিত	VM	অজানা	JJ	দুর্ভাগ্যজনকভাবে	RB
অন্যায়	NN	অবিশ্বাস	VM	অজ্ঞাত	JJ	দুর্ভাগ্যবশত	RB
অনুতাপ	NN	অশ্রদ্ধা	VM	অজ্ঞান	JJ	বিষণ্ণভাবে	RB
অন্তর্জ্বালা	NN	অসম্মান	VM	অযোগ্য	JJ	লজ্জাকরভাবে	RB
অন্তর্বেদনা	NN	আঘাত	VM	অযথা	JJ	শোকজনকভাবে	RB
অনুশোক	NN	আতঙ্কিত	VM	অযথার্থ	JJ		
অপকার	NN	আশঙ্কা	VM	অতিষ্ঠ	JJ		
অপকৃতি	NN	আহত	VM	অতৃপ্ত	JJ		
অপকর্ম	NN	কাঁদছে	VM	অধার্মিক	JJ		
অপব্যবহার	NN	কাদলো	VM	অধরা	JJ		
অপ্রেম	NN	কান্দুম	VM	অধরাই	JJ		
অমানুষ	NN	কান্না	VM	অনাস্থাভাজন	JJ		
অমানুষের	NN	ক্লেশ	VM	অনিষ্টকর	JJ		
অশান্তি	NN	কষ্ট	VM	অনুচিত	JJ		
অশান্তিতে	NN	কষ্টভোগ	VM	অননুকূল	JJ		
অশ্রদ্ধা	NN	ক্ষতি	VM	অনুপযুক্ত	JJ		
অসত্যবাদী	NN	খিটখিট	VM	অনুপযোগী	JJ		
অসদাচার	NN	খিটিমিটি	VM	অনুরাগহীন	JJ		
অসন্তোষ	NN	খিটমিট	VM	অনর্থক	JJ		
অসন্তুষ্টি	NN	খেপা	VM	অপকারক	JJ		
অসুবিধা	NN	গালাগালি	VM	অপকারী	JJ		
অস্বস্তি	NN	গালি	VM	অপকৃত	JJ		
অসম্মান	NN	গালিগালাজ	VM	অপকৃষ্ট	JJ		
অসহায়তা	NN	ঘৃণা	VM	অপরিচিত	JJ		
অহিত	NN	চোট	VM	অপরিজ্ঞাত	JJ		
আক্ষেপ	NN	জ্বালাতন	VM	অপ্রীতিকর	JJ		
আঘাত	NN	ঝামেলায়	VM	অপ্রসন্ন	JJ		
আঘাতে	NN	ঝঞ্ঝাটে	VM	অবিজ্ঞাত	JJ		
আজাইরা	NN	যন্ত্রণা	VM	অবিশ্বস্ত	JJ		
আপদ	NN	দুঃখ	VM	অব্যবহার্য	JJ		
আফসোস	NN	দুঃখে	VM	অভিমানী	JJ		
আশঙ্কা	NN	দুশ্চিন্তায়	VM	অভদ্র	JJ		
আশাভঙ্গ	NN	পীড়িত	VM	অর্থহীন	JJ		
উদাসীনতা	NN	পীড়া	VM	অলস	JJ		
ঔদাসীন্য	NN	পোহান	VM	অশান্তিকর	JJ		
ঔদাস্য	NN	প্রতারিত	VM	অশালীন	JJ		
কায়ক্লেশ	NN	বকা	VM	অশিক্ষিত	JJ		

কান্না	NN	বকাবকি	VM	অশোভন	JJ		
কান্নাকাটি	NN	বিজড়িত	VM	অশুভ	JJ		
কান্নার	NN	বিপদে	VM	অশ্রদ্ধ	JJ		
ক্রোধ	NN	বিব্রত	VM	অশ্রদ্ধাজনক	JJ		
কলঙ্ক	NN	বিভ্রান্ত	VM	অশ্রদ্ধেয়	JJ		
ক্লেশ	NN	বিরক্ত	VM	অশ্লীল	JJ		
কষ্ট	NN	বেদনা	VM	অসুখকর	JJ		
কষ্টও	NN	বরদাস্ত	VM	অসুখী	JJ		
কষ্টগুলি	NN	বহন	VM	অসৎ	JJ		
কষ্টগুলো	NN	ভুগা	VM	অসত্য	JJ		
কষ্টে	NN	ভোগা	VM	অসন্তোষজনক	JJ		
কষ্টের	NN	রাগানো	VM	অসুন্দর	JJ		
কষ্টভোগ	NN	লোকসান	VM	অসুবিধাজনক	JJ		
ক্ষত	NN	লজ্জাবোধ	VM	অস্বচ্ছন্দ	JJ		
ক্ষতি	NN	লজ্জিত	VM	অস্বস্তিকর	JJ		
খেদ	NN	সন্দিহান	VM	অস্বস্তিপূর্ণ	JJ		
খারাপ	NN	সহা	VM	অসুস্থ	JJ		
খরাপ	NN	হিংসা	VM	অসহায়	JJ		
গরমি	NN			অসহ্য	JJ		
ঘাত	NN			অসহনীয়	JJ		
চাপ	NN			অহিতকামী	JJ		
চাপে	NN			অহিতকর	JJ		
চোট	NN			আজাইরা	JJ		
জখম	NN			আত্নদর্পী	JJ		
জ্বালা	NN			আত্মাভিমানী	JJ		
জালা	NN			আবাল	JJ		
জ্বালায়	NN			আশঙ্কিত	JJ		
ঝুঁকি	NN			আশাবিহীন	JJ		
ঝামেলা	NN			আশাশূন্য	JJ		
যন্ত্রণা	NN			আশাহীন	JJ		
যন্ত্রনা	NN			আশাহত	JJ		
তকলিফ	NN			আস্থাহীন	JJ		
তীব্র	NN			উগ্র	JJ		
ত্রুটি	NN			উচ্ছৃঙ্খল	JJ		
দুঃখ	NN			উদ্দেশ্যহীন	JJ		
দু:খের	NN			উদ্বিগ্ন	JJ		
দুঃখই	NN			উদ্বেগপূর্ণ	JJ		
দুঃখে	NN			উদ্ভট	JJ		
দুঃখদুর্দশা	NN			কাতর	JJ		
দুঃসময়	NN			কঠিন	JJ		
দুঃসময়ে	NN			কুৎসিত	JJ		
দূরসময়	NN			কদাকার	JJ		
দূরসময়ে	NN			কদর্য	JJ		
দেমাক	NN			ক্রুটিপূর্ণ	JJ		
দোষ	NN			করুণ	JJ		
দজ্জাল	NN			করুণযোগ্য	JJ		
দুর্দশা	NN			ক্রুদ্ধ	JJ		

দুর্ব্যবহার	NN			ক্লেশকর	JJ		
দুর্বৃত্ত	NN			ক্লেশজনক	JJ		
দুর্ব্রিত্ত	NN			ক্লেশদায়ক	JJ		
দুষ্কৃত	NN			কুশ্রী	JJ		
দুষ্কর্ম	NN			কষ্টকর	JJ		
নাশকতা	NN			ক্ষতিকারক	JJ		
নাশকতার	NN			ক্ষতিকর	JJ		
নির্বুদ্ধিতা	NN			ক্ষতিজনক	JJ		
নির্বুদ্ধিতার	NN			ক্ষমতালোভী	JJ		
পীড়া	NN			কষষ্টদায়ক	JJ		
পরিতাপ	NN			কড়া	JJ		
প্রতারক	NN			খেলো	JJ		
প্রমাদ	NN			খারাপ	JJ		
প্রহার	NN			খরাপ	JJ		
ফালতু	NN			গোমড়া	JJ		
বিনাশ	NN			গোঁয়ারগোবিন্দ	JJ		
বিপত্তি	NN			গতিহীন	JJ		
বিপদ	NN			গরম	JJ		
বিরক্ত	NN			ঘৃণার্হ	JJ		
বেচারা	NN			ঘৃণিত	JJ		
বেচারার	NN			ঘৃণ্য	JJ		
বেদনা	NN			চিন্তানিমগ্ন	JJ		
ব্যথা	NN			চিন্তান্বিত	JJ		
ব্যাথা	NN			চিন্তিত	JJ		
ব্যথায়	NN			চিন্তিতবোধ	JJ		
বদমাশ	NN			জঘন্য	JJ		
ভুগানি	NN			জগন্য	JJ		
ভ্রম	NN			জালাতুনে	JJ		
ভুল	NN			জেদি	JJ		
ভয়	NN			জটিল	JJ		
মিথ্যাবাদী	NN			জ্বালাতনকর	JJ		
মন্দ	NN			ঝুট	JJ		
মর্মবেদনা	NN			যন্ত্রণাকাতর	JJ		
মুস্কিল	NN			যন্ত্রণাকর	JJ		
রাগ	NN			যন্ত্রণাজনক	JJ		
রোষ	NN			যন্ত্রণাদগ্ধ	JJ		
লোকসান	NN			যন্ত্রণাদায়ক	JJ		
শঙ্কা	NN			যন্ত্রণাময়	JJ		
শোক	NN			যন্ত্রনাময়	JJ		
শূন্যতা	NN			তুচ্ছ	JJ		
শূন্যতার	NN			তটস্থ	JJ		
সংকট	NN			দুঃখকর	JJ		
সন্তাপ	NN			দুঃখার্ত	JJ		
সমস্যা	NN			দুঃখিত	JJ		
সমসা	NN			দুঃখজনক	JJ		
সমস্যাটা	NN			দুঃখদ	JJ		
সমস্যাটার	NN			দুঃখদায়ক	JJ		

সর্বনাশ	NN			দুঃখদায়ী	JJ		
সর্বস্বান্ত	NN			দুঃখপ্রদ	JJ		
সহন	NN			দুঃখময়	JJ		
হানি	NN			দুঃশাসন	JJ		
হিংসা	NN			দুঃসাহসী	JJ		
				দুঃস্থ	JJ		
				দজ্জাল	JJ		
				দুর্গত	JJ		
				দুর্দশাগ্রস্ত	JJ		
				দুর্নীতিপরায়ণ	JJ		
				দুর্বোধ্য	JJ		
				দুর্বৃত্ত	JJ		
				দুর্বিত্ত	JJ		
				দুর্বল	JJ		
				দুর্ভাগ্য	JJ		
				দুর্ভাগ্যজনক	JJ		
				দুরূহ	JJ		
				দুশ্চিন্তিত	JJ		
				দুশ্চরিত্র	JJ		
				দূষিত	JJ		
				দুষ্ট	JJ		
				দুষ্টু	JJ		
				নিঃস্ব	JJ		
				নিঃসহায়	JJ		
				নিকৃষ্ট	JJ		
				নিদ্রি	JJ		
				নিরানন্দ	JJ		
				নিরালয়	JJ		
				নিরাশ	JJ		
				নিরাশ্রয়	JJ		
				নিরুপায়	JJ		
				নিরবলম্ব	JJ		
				নিরবলম্বন	JJ		
				নির্মম	JJ		
				নিরর্থক	JJ		
				নিষ্ক্রিয়	JJ		
				নিষ্ঠুর	JJ		
				নিষ্প্রাণ	JJ		
				নিষ্ফল	JJ		
				নীচ	JJ		
				নোংরা	JJ		
				ন্যায়পরায়ণতাহীন	JJ		
				নষ্ট	JJ		
				পাপিষ্ঠ	JJ		
				পাপী	JJ		
				পাপপূর্ণ	JJ		
				পীড়াদায়ক	JJ		

				পীড়িত	JJ		
				প্রতারিত	JJ		
				ফালতু	JJ		
				বাজে	JJ		
				বিকটদর্শন	JJ		
				বিকল	JJ		
				বিঘ্নজনক	JJ		
				বিদখুটে	JJ		
				বিদঘুটে	JJ		
				বিদ্বেষপূর্ণ	JJ		
				বিপর্যস্ত	JJ		
				বিফল	JJ		
				বিমনা	JJ		
				বিমর্শ	JJ		
				বিমর্ষ	JJ		
				বিরক্তিকর	JJ		
				বিরক্তিজনক	JJ		
				বিশ্রী	JJ		
				বিষাদিত	JJ		
				বিষণ্ণ	JJ		
				বিড়ম্বনাকর	JJ		
				বীভৎস	JJ		
				বেইমান	JJ		
				বেকার	JJ		
				বেখাপ্পা	JJ		
				বেচারা	JJ		
				বেদনাজনক	JJ		
				বেদনাদায়ক	JJ		
				বেদনার্ত	JJ		
				বেমক্কা	JJ		
				বেমানান	JJ		
				বঞ্চিত	JJ		
				ব্যথিত	JJ		
				ব্যর্থ	JJ		
				বৃথা	JJ		
				বৃথাই	JJ		
				বদ	JJ		
				বদমাশ	JJ		
				বহাবহ	JJ		
				ভয়ঙ্কর	JJ		
				ভয়াবহ	JJ		
				মারাত্মক	JJ		
				মিথ্যা	JJ		
				মিথ্যাবাদী	JJ		
				মন্দ	JJ		
				মনমরা	JJ		
				মূর্খ	JJ		

				মর্মাহত	JJ		
				মুশকিল	JJ		
				রুক্ষ	JJ		
				রাগান্বিত	JJ		
				রাগি	JJ		
				রোগার্ত	JJ		
				রুষ্ট	JJ		
				লক্ষ্যহীন	JJ		
				শিক্ষাহীন	JJ		
				শোচনীয়	JJ		
				শ্রদ্ধাহীন	JJ		
				সাহায্যহীন	JJ		
				সহ্য	JJ		
				হারানো	JJ		
				হেয়	JJ		
				হতাশ	JJ		
				হতাশ্বাস	JJ		
				হতচেতন	JJ		
				হতভাগ্য	JJ		
				হৃদয়হীন	JJ		

Table 7**:** Bangla negative polarity word list. NN, VM, JJ and RB stand for Noun, Verb, Adjective and Adverb respectively. Highlighted words in gray indicate to the inflected and misspelled forms.

REFERENCES

[1] Twitter: https://twitter.com/

[2] H. Kanayama and T. Nasukawa. Fully automatic lexicon expansion for domain-oriented sentiment analysis. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Sydney, Australia, 2006.

[3] N. Kobayashi, K. Inui, K. Tateishi, and T. Fukushima. Collecting evaluative expressions for opinion extraction. In Proceedings of IJCNLP 2004, pages 596–605, 2004.

[4] Y. Suzuki, H. Takamura, and M. Okumura. Application of semi-supervised learning to evaluative expression classification. In Proceedings of the 7th International Conference on Intelligent Text Processing and Computational Linguistics, 2006.

[5] H. Takamura, T. Inui, and M. Okumura. Latent variable models for semantic orientations of phrases. In Proceedings of the 11th Meeting of the European Chapter of the Association for Computational Linguistics, 2006.

[6] Y. Hu, J. Duan, X. Chen, B. Pei, and R. Lu. A new method for sentiment classification in text retrieval. In IJCNLP, pages 1–9, 2005.

[7] T. Zagibalov and J. Carroll. Automatic seed word selection for unsupervised sentiment classification of chinese text. In Proceedings of the Conference on Computational Linguistics, 2008.

[8] S.M. Kim and E. Hovy. Identifying and analyzing judgment opinions. In Proceedings of the Human Language Technology Conference - North American chapter of the Association for Computational Linguistics, New York City, NY, 2006.

[9] R. Mihalcea, C. Banea, and J. Wiebe. Learning multilingual subjective language via cross-lingual projections. In Proceedings of the Association for Computational Linguistics, Prague, Czech Republic, 2007.

[10] C. Banea, R. Mihalcea, J. Wiebe, and S. Hassan. Multilingual subjectivity analysis using machine translation. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2008), Honolulu, Hawaii, 2008.

[11] A. Das and S. Bandyopadhyay (2009a). Subjectivity Detection in English and Bengali: A CRF-based Approach., In Proceeding of ICON 2009, December 14th-17th, 2009, Hyderabad.

[12] D Das, S Bandyopadhyay. Labeling emotion in Bengali blog corpus–a fine grained tagging at sentence level, In Proceedings of the 8th Workshop on Asian Language Resources, pages 47–55, Beijing, China, August 2010.

[13] Wikipedia: www.wikipedia.org/

[14] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? Sentiment classification using machine learning techniques. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 79-86, 2002.

[15] Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. Technical report, Stanford.

[16] Pak, A., and Paroubek, P. 2010 (May). Twitter as a corpus for sentiment analysis and opinion mining. In N. C. C. Chair, K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, and D. Tapias (eds.), Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC'10), Valletta, Malta; ELRA, pp.19–21. European Language Resources Association.

[17] Davidov, D., Tsur, O., and Rappoport, A. 2010a. Enhanced sentiment learning using Twitter hashtags and smileys. In Proceedings of the 23rd International Conference on Computational Linguistics: Posters, COLING '10, pp. 241–9. Stroudsburg, PA: Association for Computational Linguistics.

[18] Luciano Barbosa and Junlan Feng. 2010. Robust sentiment detection on twitter from biased and noisy data. Proceedings of the 23rd International Conference on Computational Linguistics: Posters, pages 36–44.

[19**]** ANEW: http://csea.phhp.ufl.edu/media/anewmessage.html

[20] General Inquirer: http://www.wjh.harvard.edu/~inquirer/

[21] Opinion Finder: http://mpqa.cs.pitt.edu/opinionfinder/

[22] SentiWordNet: http://sentiwordnet.isti.cnr.it/

[23] WordNet-Affect: http://wndomains.fbk.eu/wnaffect.html

[24] Turney, P. D. 2002. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL ’02), pp. 417–24. Stroudsburg, PA: Association for Computational Linguistics.

[25] Maynard, D., and Funk, A. 2012. Automatic detection of political opinions in tweets. In R. Garcia-Castro, D. Fensel, and Antoniou, G. (eds.), The Semantic Web: ESWC 2011 Workshops, Lecture Notes in Computer Science, Vol. 7117, pp. 88–99. Berlin/Heidelberg: Springer.

[26] FA Nielsen. 2011. A new ANEW: Evaluation of a word list for sentiment analysis in microblogs.

[27] A Kennedy, D Inkpen. 2006. Sentiment classification of movie reviews using contextual valence shifters. In Computational Intelligence, Wiley Online Library

[28] Xiaojun Wan. 2008. Using Bilingual Knowledge and Ensemble Techniques for Unsupervised Chinese Sentiment Analysis. In Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, pages 553–561, Honolulu.

[29] http://leebecker.com/resources/semeval-2013/

[30] V. Hatzivassiloglou and K. McKeown, Predicting the semantic orientation of adjectives. In Proceedings of the Joint ACL/EACL Conference,2004, pp. 174–181

[31] A. Kumar. and T. M. Sebastian, Sentiment Analysis on Twitter, International Journal of Computer Science (IJCSI), Vol. 9, Issue 4, No 3, July 2012

[32] Ovidhan: http://ovidhan.org/

[33] Samsad: http://dsal.uchicago.edu/dictionaries/biswas-bengali/

[34] Bangla Wordnet of Indian Stastical Institute: http://www.isical.ac.in/~lru/wordnetnew/index.php/site/aboutus

[35] https://dev.twitter.com/docs/api/1.1

[36] http://nltk.org/api/nltk.tokenize.html

[37] http://nltk.org/api/nltk.tag.html

[38] https://github.com/ankur-india/bangla_pos_tagger

[39] Xiaojin Zhu. 2005. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison. http://www.cs.wisc.edu/jerryzhu/pub/ssl survey.pdf.

[40] Matthias Seeger. 2001. Learning with labeled and unlabeled data. Technical report, University of Edinburgh.

[41] Rada Mihalcea. 2004. Co-training and Self-training for Word Sense Disambiguation.

[42] http://sentiment.christopherpotts.net/code-data/happyfuntokenizing.py

[43] http://ltrc.iiit.ac.in/analyzer/bengali

[44] Eugenio Martínezcámara, M. Teresa Martínvaldivia,L. Alfonso Ureñalópez and A Rturo Montejoráez. Sentiment analysis in Twitter. Natural Language Engineering.

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

[46] Kamal Nigam, John Lafferty, and Andrew McCallum. 1999. Using maximum entropy for text classication. In Proc. of the IJCAI-99 Workshop on Machine Learning for Information Filtering, pages 61-67.

[47] http://nltk.org/_modules/nltk/classify/maxent.html

[48] Jacob Perkins. 2010. Python Text Processing with NLTK 2.0 Cookbook.

[49] http://nltk.org/api/nltk.metrics.html

[50] Preslav Nakov, Sara Rosenthal, Zornitsa Kozareva, Alan Ritter, Theresa Wilson. 2013. Sentiment Analysis in Twitter.

[51] E Kouloumpis, T Wilson, J Moore. Twitter sentiment analysis:
The Good the Bad and the OMG!. ICWSM, 2011.

[52] A Bermingham, AF Smeaton. Classifying sentiment in microblogs:
is brevity an advantage?. CIKM, 2010.

[53] Zhang, L., Ghosh, R., Dekhil, M., Hsu, M., and Liu, B. 2011. Combining lexicon-based and learning-based methods for Twitter sentiment analysis. Technical Report HPL-2011-89.

[54] Dasgupta, Sajib and Vincent Ng. 2009.Mine the easy, classify the hard: A semi-supervised approach to automatic sentiment classification. In *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics*, pages 701–709,Singapore.

[55] Li, Shoushan, Chu-Ren Huang, Guodong Zhou, and Sophia Yat Mei Lee. 2010. Employing personal/impersonal views in supervised and semi-supervised sentiment classification. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 414–423, Uppsala.

[56] http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html#lexicon