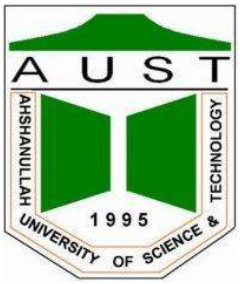


Ahsanullah University of Science & Technology

Department of Computer Science & Engineering

SENTIMENT ANALYSIS FOR TEXT DOCOUMENT IN BENGALI

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

Bachelor of Science in Computer Science & Engineering

By

Tazim Haque

ID: 11.01.04.055

Md.Rifat-Ut-Tauwab

ID: 11.01.04.038

Tazreen Bhuiyan

ID: 11.01.04.075

Azmin Ahmed

ID: 11.01.04.043

DECLARATION

We herewith proclaim that the thesis is based on the outcome experimented and discovered by us. All the stuff used in the thesis is mentioned in the reference. Neither any part nor whole of this report is submitted before for any degree.

CERTIFICATE

This is to certify that the thesis named **"Sentiment Analysis for Text Document in Bengali"** is written by **Tazim Haque, Md.Rifat-Ut-Tauwab** ,**Tazreen Bhuiyan** and **Azmin Ahmed** under the direction of their supervisor and approved by all the members of thesis committee and approved by the Head of the Department of Computer Science & Engineering in partial fulfillment of the requirements of the degree of Bachelor of Science in Computer Science & Engineering.

_____ _____

(Project Supervisor) Head of the Department

Department of CSE

ACKNOWLEDGEMENTS

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

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Sentiment analysis or Opinion mining refers to the application of natural language processing, computational linguistics and text analytics to identify and extract sentimental information from text. A basic task in sentiment analysis is classifying the polarity of a given text at the document, sentence or features – whether the expressed sentiment in a document, sentence or an entity features is positive (happy), negative (sad) and neutral (memorable). The feature level of sentiment demands proper structures for more precise sentiment extraction. Sentiment Analysis from natural language text is multidisciplinary problem. Sentiment Analysis defines an overall problem, which address multiple perspectives of sub-problems. Human sentiment knowledge grows with its age and daily cognitive interactions. Therefore an intelligent human should need some prior language to act properly. Sentiment knowledge acquisition is generally wrapped into a lexicon called Sentimental Lexicon which is similar to classical pattern classification problems. But for the proper analysis, proper structurization is must. It involves the perfect identification of sentiment topic. Therefore we need to develop such a system that should meet the satisfaction level of the end level users. A perfect summarization of a sentiment is the first priority for us. The thesis reported here is experimented in both Bengali and English. But our main concern is of course Bengali. Because most of the work related sentiment analysis are found in English. English is in the third position among the most widely used languages in the world[1]. So, construction of resources and tools for sentiment analysis in languages other than English is a growing need. Moreover, we are in the internet era. Microblog and social networking sites are like our daily friend. Microblog posts and social networking sites statuses and comments are not just posted in English but in other languages as well. In many cases we find a mixture of more than one language. We are working on Bangla. We find it necessary because it is one of the most spoken languages, ranked seventh most spoken language in the world[2]. We aim to extract the sentiments of the Bangla language users and then identify the overall porlarity of texts as either positive or negative. We have chosen Prothom-Alo, a famous online newspaper as our site[3]. Because it the number one online portal in our country[4]. We can get huge posts and comments from this site and the best part is every day we are getting new news, posts and comments which help us to experiment and analysis our discovers.

The need of the end user is the main motto behind our research. The outcome should lead to a development of real time sentiment analysis system and which will successfully satisfy the need or requirements of the end user. Let us have a look at some real life needs if the end user. For example, a market surveyor from company A may identify the need to find out the changes in public opinion about their product X after release of product Y by another company B. the different aspects of product Y that the public consider better than X are also points of interest. These aspects typically may be durability of the product, power options, weight, colors and many other issues that depend on the particular product. Here the end user in not only looking for the binary positive or negative result but also they are more interested in specific sentiment

analysis. Only sentiment detection and classification is not enough to satisfy the need of the end user. The system should be capable enough to understand and extract out the specific sentiments hidden in the natural language texts. It should be capable enough that end user can get the exact sentiment of all combination of natural texts. Texts may be in combination of more than one language, may be emoticon, numbers etc.

1. What is Sentiment Analysis

Before any scientific research, researchers need to know the proper definitions of the problems in order to solve it. So, the most important question raised before we started working is “What is Sentiment Analysis?” But the answer cannot be given in one or two words. Man researchers attempted to answer the question psychology, philosophy, psycholinguistics and even cognitive science. The researchers attempted to give their own definitions. Among those research endeavors, the General Inquires (Stone, 1966) System and the Subjectivity definition by Janyce Wiebe (Wiebe et. al., 1990) are the milestones that mark the avenue to the current research trend of today. Many definitions are discovered now those describe the main logic strongly.

Sentiment analysis also known as opinion mining refers to the use of natural language processing, text analysis and computational linguistics to identify and extract subjective information in source materials[16].

Generally speaking, sentiment analysis aims to determine the attitude of a speaker or a writer with respect to some topic or the overall contextual polarity of a document. The attitude may be his or her judgment or evaluation, affective state (that is to say, the emotional state of the author when writing), or the intended emotional communication (that is to say, the emotional effect the author wishes to have on the reader).

Sentiment Analysis/Opinion Mining from natural language text is a multifaceted and multidisciplinary AI problem. It tries to narrow the communicative gap between the highly sentimental human and the
sentimentally challenged computers by developing computational systems that can recognize and respond to the sentimental states of the human users. There is a perpetual debate about the better ways of collecting intelligence either by following the functional path of biological human intelligence or generating new methodologies for completely heterogeneous mechatronics machine and redefine a
completely new horizon called electronic intelligence. The research endeavors in the present task is to find out the optimum solution strategies for machines that either mimic the techniques of self-organized biological human intelligence or at least can simulate the functional similarities of human sentimental intelligence.

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 online portal sites, more specifically Prothom-Alo, as we only deal with Prothom-Alo corpus, we perform sentiment analysis of their posts.

Much of the research work on polarity classification of Microblog or news portals posts have 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 and portals are not just posted in English, but in other natural languages as well. Work on other languages is growing, including Japanese ([5], [6], [7], [8]), Chinese ([9], [10]), German [11], and Romanian ([12], [13]). Much of the work on sentiment analysis for Bangla (or Bengali) language has been applied to the news corpus and blogs ([14],[15]), but we could not find few research paper which focus on the issue of extracting user opinions and views from Bangla online portals.

2. **Related Work in English:**

We briefly outline the main theme of the research carried out on the English language. There are a large number of approaches that has been developed for classifying sentiments in English. These methods can be classified into two categories-

a. machine learning or statistical-based approach and

b. 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. [17] 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 [18].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%. [19] follow the same procedures as [18] 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 [20] use 50 hashtags and 15 emoticons as sentiment labels to train a supervised sentiment classifier using the K Nearest Neighbors (KNN) algorithm. In [21], 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 [22], General Inquirer [23], OpinionFinder [24], SentiWordNet [25] and WordNet-Affect [26]. One of the initial works to apply unsupervised techniques to sentiment classification problem is by [27]. 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 [28], 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 [29] and [30]. 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 [31]. 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 [32], [33] and [34].

There are also some related works on English in sentiment summarization. May be it is not exactly related but is also a part of natural language processing.

Many previous works on extractive summarization use two major steps: (1) ranking the sentences based on their scores which are computed by combining few or all of the features such as term frequency (TF), positional information and cue phrases (Baxendale, 1958; Edmundson, 1969; Luhn, 1958; Lin and Hovy 1997) and (2) selecting few top ranked sentences to form an extract. The very first work on automatic text summarization by Luhn (1958) computes salient sentences based on word frequency (number of times a word occurs in a document) and phrase frequency. Although subsequent research has developed sophisticated summarization methods based on various new features, the work presented by Edmundson (1969) is still followed today as the foundation for extraction based summarization.

Barzilay and Elhadad (1997) described a summarization approach that used ***lexical chaining*** method to compute the salience of a sentence. Cohesion (Halliday and Hasan, 1976) is a method for sticking together different parts of the text. ***Lexical cohesion*** is the simplest form of cohesion. Lexical Cohesion links the different parts of the text through semantically related terms, co-reference, ellipsis and conjunctions. Lexical cohesion also involves relations such as reiteration, synonymy, hypernymy (IS-A relations such as "dog-is-a-kind-of-animal", "wrist-is-a-part-of-hand"). The concept of lexical chain was introduced in (Morris and Hirst, 1991). They characterized lexical chain as a sequence of related words that spans a topical unit of text. In other words, lexical chain is basically lexical cohesion that occurs between two terms and among sequences of related words. Barzilay and Elhadad (1997) used a WordNet (Miller, 1995) to construct the lexical chains.

Dorr et al. (2003) developed the Hedge Trimmer that uses a parse-and-trim based approach to generate headlines. In this approach, the first sentence of a document is parsed using a parser and then the parsed sentence is compressed to form a headline by eliminating the unimportant constituents of the sentence using a set of linguistically motivated rules.

TOPIARY (Zajic et al., 2004), a headline generation system, combines the compressed version of the lead sentence and a set of topic descriptors generated from the corpus to form a headline. The sentence is compressed using the approach similar to the approach in (Dorr et al. 2003) and the topic descriptors. A number of approaches for creating abstracts have been conceptualized without much emphasis on the issue that a true abstract may contain some information not contained in the document. Creating such an abstract requires external information of some kind such as ontology, knowledge base etc. Since large-scale resources of this kind are difficult to develop, abstractive summarization has not progressed beyond the proof-of-concept stage.

3. **Related Works in Bangla:**

We briefly outline the main theme of the research carried out on the Bangla language. But a very limited numbers of research are found in Bangla. We hardly found two researches working in Bangla.

The first work we are discussing is about the text summarization. The main objective of the work presented in their paper is to generate an extract from a Bengali document. They have followed a simple and easy-to-implement approach to Bengali single document text summarization because the sophisticated summarization system requires resources for deeper semantic analysis. Bengali is a resource constrained language and NLP (natural language processing) research activities on Bengali have recently been started. In their work presented in this paper, they have investigated the impact of thematic term feature and position feature on Bengali text summarization. To their knowledge, no generic text summarization system for Bengali is available for comparison to their system. So, they have compared the proposed method to the LEAD baseline which was defined for single document text summarization task in past two DUC conferences DUC 2001 and DUC 2002. LEAD baseline considers the first *n* words of an input article as a summary, where *n* is a predefined summary length. The proposed summarization method is extraction based. It has three major steps: (1) preprocessing (2) sentence ranking (3) summary generation. The preprocessing step includes stop-word removal, stemming and breaking the input document in to a collection of sentences. Using stemming, a word is split into its stem and affix. The design of a stemmer is language specific, and requires some significant linguistic expertise in the language. A typical simple stemmer algorithm involves removing suffixes using a list of frequent suffixes, while a more complex one would use morphological knowledge to derive a stem from the words. After an input document is formatted and stemmed, the document is broken into a collection of sentences and the sentences are ranked based on two important features: thematic term and position. They consider length of a sentence as a feature because they observe that if a sentence is too short, but it occurs in the beginning paragraph of a document it is sometimes selected due to its positional advantage. On the other hand, if a sentence is too long, it is sometimes selected due to the fact that it contains many words. So, they eliminate the sentences which are too short or too long. A summary is produced after ranking the sentences based on their scores and selecting K-top ranked sentences, when the value of K is set by the user. To increase the readability of the summary, the sentences in the summary are reordered based on their appearances in the original text.

And the next we will discuss about Das and Bandyopadhay's contribution to the community of opinion extraction. Resource acquisition is one of the most challenging obstacles to work with resource constrained languages like Bengali. Extensive NLP research activities in Bengali have started recently but resources like annotated corpus, various linguistic tools are still unavailable for Bengali in the required measure. For the present task a Bengali news corpus has been developed from the archive of a leading Bengali news paper available on the Web

(http://www.anandabazar.com/). A portion of the corpus from the editorial pages, i.e., Reader's opinion section or Letters to the Editor Section containing 28K word forms has been manually annotated with sentence level subjectivity and discourse level theme words. Detailed reports about this news corpus development in Bengali can be found in (Das and Bandyopadhyay,2009b). From the collected document set (Letters to the Editor Section), some documents have been chosen for the annotation task. Some statistics about the Bengali news corpus is represented in the Table 1. Documents that have appeared within an interval of four months are chosen on the hypothesis that these letters to the editors will be on related events. A simple annotation tool has been designed for annotating the sentences considered to be important for opinion summarization. Three annotators (Mr. X, Mr. Y and Mr. Z) participated in the present task.

The annotation tool highlights the sentiment words (Das and Bandyopadhyay, 2010a) by four different colors within a document according to their POS categories (Noun, Adjective, Adverb and Verb). This technique helps to increase the speed of annotation process. Finally 100 annotated documents have been developed. The agreement of annotations among three annotators has been evaluated. The agreements of tag values at theme words level and sentence levels are listed in Tables 2 and 3 respectively

	NEWS	*BLOG*
Total number of documents	100	-
Total number of sentences	2234	300
Avgerage number of sentences in a	22	-
Total number of wordforms	28807	4675
Avgerage number of wordforms	288	-
Total number of distinct wordforms	17176	1235

Table 1: Statistics of Bengali Corpus, used to measure the Coverage of the developed SentiWordNet(Bengali)

Annotators	*X vs Y*	*X vs Z*	*Y vs Z*	*Avg.*
Percentage	82.64%	71.78%	80.47%	78.30%
All Agree		69.06%		

Table 2: Agreement of annotators at theme words level

Annotators	X vs Y	X vs Z	Y vs Z	Avg.
Percentage	73.87%	69.06%	60.44%	67.80%
All Agree		58.66%		

Table 3: Agreement of annotators at theme sentence level

From the analysis of inter-annotator agreement, it is observed that the agreement drops fast as the number of annotator's increases. It is less possible to have consistent annotations when more annotators are involved. In the present task the inter-annotator agreement is better for theme words annotation rather than candidate sentence identification for summary though a small number of documents have been considered. The set of features used in the present task have been categorized as Lexico-Syntactic, Syntactic and Discourse level features. These are listed in the Table 4 below and have been described in the subsequent subsections.

Types	*Features*
Lexico-Syntactic	POS
	SentiWortdNet
	Frequency
	Stemming
Syntactic	Chunk Level
	Dependency Parsing Depth
Discourse Level	Title of the Document
	First Paragraph
	Term distribution
	Collocation

Table 4: Features

Term Frequency (TF) plays a crucial role to identify document relevance in Topic-Based Information Retrieval. The motivation behind developing

Theme detection technique is that in many documents relevant words may not occur frequently or irrelevant words may occur frequently. Present system is an extractive opinion summarization system for Bengali. But identifying those clusters is not only a step toward generating document level opinionated news summary rather another major step is to extract thematic sentences from each theme cluster that reflects the contextual concise content of the current theme cluster. Extraction of sentences based on their importance in representing the shared subtopic (cluster) is an important issue and it regulates the quality of the output summary. They have used Information Retrieval (IR) based technique to identify the most "informed" sentences from any cluster and it can be termed as IR based cluster center for that particular cluster. With the adaptation of ideas from page rank algorithms (Page et al., 1998), it

can be easily observed that a text fragment (sentence) in a document is relevant if it is highly related to many relevant text fragments of other documents in the same cluster. They computed the relevance of a node/sentence by summing up the edge scores of those edges connecting the node with other nodes in the same cluster. Then the nodes are given rank according to their calculated relevance scores and the top ranking sentences is selected as the candidate sentence representing the opinion summary. For example four such candidate sentences are shown in Table 5. The words in bold are the theme words based on those theme words the sentences are extracted.

Candidate Sentence	IR Score
মুহম্মদ আমিনের মত পলিটব্যুরোর নবীনতম সদস্যকেও কিন্তু বয়সের দিক থেকে নবীন ভাবা কঠিন	151
এবার চিন্তা আরো বেশি , কারন এই মূল্যবৃদ্ধির পিছনে যেমন দেশের ভিতরের জিনিসপত্রের যোগান কমে যাওয়া আছে,তেমনি আছে আন্তর্জাতিক বাজারে মূল্যবৃদ্ধির প্রবণতা	167
স্বাধীনতার পর এত বছর কেটে গেল, এখনো প্রায় সকল সরকারী পরিকল্পনার পিছনে একটি ভাবাদর্শই কাজ করে, যেনতেন প্রকারে দলীয় স্থিতি নিশ্চিত করা	130

Table 5: Candidate sentences

And the last work related on Bengali is the work done by Saika Chowdhury and Wasifa Chowdhury. For their work, they choose Twitter as the microblogging site as it is one of the most popular microblogging platforms in the world. In order to create a Bangla sentiment lexicon, which contains Bangla words annotated with their corresponding polarity (positive/negative) and Part-of-Speech (POS), they 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. They use this Bangla polarity lexicon for the rule-based classifier and feature extraction. Their dataset is a collection of Bangla tweets downloaded by querying Twitter REST API v1.1 [46] 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, they collected a total of 1300 tweets by polling Twitter API. They split their dataset into training set and test set, comprising 1000 and 300 tweets respectively. The raw tweets data obtained through Twitter API are noisy and hence, are preprocessed. They perform pre-processing through three steps – tokenization, normalization, and POS Tagging. For tokenization, NLTK's Tokenizer [47] package is used. Username (e.g., @user), URL link, hashtag (e.g., #রাগ (#angry)), retweets (e.g., RT) are

removed from the dataset. English punctuations (e.g.,'!','.','?',',',') are also removed. Twitter users use emoticons (e.g., :), :(, ;)) to express sentiment; they 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 [48]. For Bangla, POS Tagging is performed by the Bangla Pos-Tagger Package [49]. 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, they provided manually pos-tagged tweets, labeled using Bengali Shallow Parser [50], as the initial pos-tagger. As a result, Bangla tokens are pos-tagged more accurately. For training set semi-supervised process is usedSelf-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 [51]. To make this rule-based classifier, they set the following rules, keeping in mind that tweets are short (restricted to 140 characters):

$$1.\ \text{If } \text{count}_{\text{positive}} > \text{count}_{\text{negative}}:$$

Label 'positive'

$$2.\ \text{If } \text{count}_{\text{negative}} > \text{count}_{\text{positive}}:$$

Label 'negative'

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. Word N-gram, Emoticon, Lexicon, Pos-tagging and Negation are used for feature extraction.
Each tweet is represented as a contiguous sequence of N tokens called an N-gram. We use unigrams and bigrams for their work. Emoticons are the use of letters and symbols to convey facial expressions. The regular expression used to extract emoticons from tweets during preprocessing is adopted from Christopher Potts' tokenizing script [52]. They use the emoticon polarity dictionary developed by Leebecker et al [53] as their emoticon lexicon. As their Bangla polarity lexicon and the English lexicon [54] contain strong positive and negative sentiment expressing words, they use the word entries in the lexicons as features. They 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. 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, they didn't follow the negation handling method specified in [55], where every word following the negation word is appended with a 'NEG' suffix to reverse its sentiment.

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

4. **Motivation**

Sentiment Analysis/Opinion Mining is one of the most pursued research topics in recent times. Recently, many researchers and companies have explored the area of opinion detection and analysis. With the increased number of Internet users, there is a proliferation of opinions available on the web. Not only do we read more opinions from the web, such as in daily news editorials, but also we post more opinions through mechanisms such as governmental web sites, product review sites, news group message boards, personal blogs and twitters. This phenomenon has opened the door for massive opinion collection, which has potential impact on various applications such as public opinion monitoring and product review summary systems.

Moreover in today's digital age, text is the primary medium of representing and communicating information, as evidenced by the pervasiveness of e-mails, instant messages, documents, weblogs, news, articles, homepages and printed materials. Our lives are now saturated with textual information and there is an increasing urgency to develop technologies to help us manage and make sense of the resulting information overload.

While expert systems have enjoyed some success in assisting information retrieval, data mining and natural language processing (NLP) systems, there is a growing necessity of sentiment

analysis systems that can automatically process the plethora of sentimental information available in online electronic text. The increasing social necessity is the driving force for the massive research effort on Sentiment Analysis/Opinion Mining. These are the reasons to work on sentiment analysis. Now, let's talk about why we choose Bangla.

First of all Bangla is our own language. We grow up by speaking in Bangla. We have a solely soft corner for it. It is one of the reasons behind choosing this topic as our thesis. Again, we find a lot of opportunities to work in it. Because we have seen a lot of work in English. Many researchers have given their potentiality on the sentiment analysis on English. So, we get a wide area to work as our will. We find it comparatively easy and we have the freedom to work freely. If we think about the economical motivation of our work on sentimental analysis, we have a clear vision too. Bangladesh is a growing country. Business sectors are improving day by day and there are lot of competitors. So, if a company needs to study their competitors they must know the sentiments of users about the common products of the competitors. But they are just common people. Most of them have no idea about the logic and knowledge of sentiment analysis. They just want to know the result. That is why sentiment analysis or opinion extraction in Bangla is must. And for these reasons we want to use our knowledge in this sector and contribute to the business world.

IT sectors of our country are just boosted up. Every sector is influenced by it. Our politics are not different from it. Our politicians are much dependent on us to know the percentage of their supporters. Especially before election they are eager to know the status of their support; they want to know the position about their competitors. For all these they need to analyze the sentiments of the voters and they need something unique of course. These are the things those motivate us to lead our thesis in natural language and sentiment analysis.

Moreover, our supervisor motivated us to stick on our decision. He supported us in every step. We become able to discover something new. This credit goes to our supervisor as well. Without his vision and thought it was nearly impossible to complete the thesis. We are really grateful to him for his solely motivation.

5. **Goals:**

Our main goal is to discover something that will fulfill the end user's demand of sentiment analysis in Bangla. Our motto was to discover something that will completely unique and different from others. By the grace of Almighty we successfully completed our thesis and created a unique dataset that can solve the polarity of the portals and thereby. Hopefully this utilization of our unique concept will solve the sentiments or polarity perfectly. We have some future plans also regarding this project. We have plans to make it more universal which will be discussed in Future Plan sections.

6.**Sentiment / Opinion Polarity Detection**

The polarity classification is the classic problem from where the cultivation of Sentiment Analysis (SA) has actually started. The problem of polarity classification involves sentiment/opinion classification into semantic classes such as positive, negative or neutral and/or other fine-grained emotion classes like happy, sad, anger, disgust and surprise. One of the most noteworthy earliest research works on sentiment polarity classification has been conducted by (Turney et. al., 2002) with review corpus. The semantic classes were considered as “thumbs up” or “thumbs down” for movie reviews. Motivated by different real-world applications, researchers have considered a wide range of semantic classes over a variety of different types of corpora or problem domains. The development of a fully automatic polarity classifier is still the basic requirement to meet the real life needs and the ultimate desire of the whole Sentiment Analysis research. In this chapter we will describe about the various polarity classification techniques, proposed by us. The chapter is organized as follows. There are several factors that make the automatic polarity classification a very challenging research problem. The various research attempts by several researchers who attempted to formulate the research problem and the solution for the sentiment polarity classification. The polarity classification technique has been developed for the Bengali language which is a resource poor language. Thus, acquisition of relevant resources and the development of appropriate tools is one of the important aspects of the present work. The resource acquisition process includes corpus collection and annotations. A dependency parser for Bengali has also been developed which is a necessary tool to detect syntactic sentimental semantics from text. The acquisition of relevant resources and the development of appropriate tools. The details of the syntactic polarity classification technique are described. During the error analysis of the syntactic polarity classifier, it has been observed that the performance of the polarity classifier mainly drops for the unknown or new words. A closer look at the error analysis points to questioning the two standard steps in the polarity classification class, the use of a prior polarity lexicon followed by the application of any NLP technique. A new method has been proposed which uses a lexical network based on Vector Space Model (VSM) that holds the contextual sentimental polarity. The problem of holding sentiment knowledge with context is defined as Sentiments in the present work.

6.1: **Prior Polarity Lexicon :**

The polarity classification problem started as a semantic orientation determination problem. Peter Turney and Vasileios Hatzivassiloglou are the pioneers who started the initial experimentations during early 90’s. In the year of 1997, Hatzivassiloglou identified the semantic orientation of adjectives. This is the first research attempt that provided the effective and empirical method of building sentiment lexicon. After a few years, Peter Turney came up with his revolutionary approach Thumbs Up and Thumbs Down for positive and negative review classification. Finally, the concept of prior polarity lexicon evolved and firmly established itself

with the innovation of SentiWordNet by Andera Esuli in 2004. All the present polarity classifiers follow a two step methodology. In the first step, classifiers identify the polarity of a text by using any dictionary of prior polarity lexicon and in the next step contextual polarity is disambiguated with the help of NLP techniques or any other fine-grained techniques. In this section, the fundamental works are mentioned that have established the theory of prior polarity lexicons. Prior polarity lexicon involves semantic orientation determination from a text and is a big challenging research issue itself. The various semantic orientation determination techniques suggested by various previous researchers are now discussed. (Hatzivassiloglou et. al., 1997) proposed their log-linear regression model to predict the orientation of conjoined adjectives. The log-linear regression model uses the number of constraints identified from large corpus and clusters the conjoined adjectives into finite number of groups of different orientations which are finally labeled as positive or negative. The approach relies on some linguistic features, or indicators, with semantic orientation of conjoined adjectives, syntactically co-occurred. They followed the hypothesis that the conjoined adjectives usually are of the same orientation, for example, fair and legitimate, corrupt and brutal. The system is trained on a large corpus to identify these relations to predict the semantic orientation of the conjoined adjectives that are linguistically anomalous. The situation is reversed for “but”, which usually connects two adjectives of different orientations, for example, short but good, far but comfortable. The system identifies and uses this indirect information

in the following stages:

1. All conjunctions of adjectives are extracted from the corpus along with relevant morphological relations.

2. A log-linear regression model combines information from different conjunctions to determine if each of the conjoined adjectives is of same or different orientation. The result is a graph with hypothesized same- or different-orientation links between adjectives.

3. A clustering algorithm separates the adjectives into two subsets of different orientation. It places the words of same orientation into the same subset.

The average frequencies in each group are compared and the group with the higher frequency is labeled as positive. This is one of the most important milestones for textual sentiment analysis research. The performance of the reported system is quite high. But the research endeavor is also important for other important aspects such as problem definition and formulation of several hypotheses that needs to be checked further for validity.

• The requirement of an automatic system for detecting the non-linguistic characteristics like semantic orientation of text is established although the authors have suggested the system for adjectives only.

• Syntactically co-occurred adjectives belong to the same semantic orientation group although there are some exceptional cases for the conjunction “but” and others.

The problem definition has motivated other researchers to pursue the research problem. One of the most cited research papers in the literature is written by (Turney, 2002). Turney devised an algorithm to extract Pointwise Mutual Information (PMI) for consecutive words and their semantic orientation. The experiments have been carried out on movie review corpus and thus the author referred the semantic orientations as “thumbs up” or “thumbs down” instead of positive or negative (Hatzivassiloglou et. al., 1997). The simple syntactic patterns considered for the experiments have been described in Table

First Word	Second Word	Third Word(Not Extracted)
JJ	NN or NNS	Anything
RB, RBR, or RBS	JJ	not NN nor NNS
JJ	JJ	not NN nor NNS
NN or NNS	JJ	not NN nor NNS
RB, RBR, or RBS	VB, VBD,VBN, or VBG	Anything

Table 6: Syntactic patterns of POS tags for pointwise mutual information (PMI) calculation (Turney , 2002)

The Brill POS Tagger (Brill, 1994)[35] tagger has been used for the task. Phrases containing words with adjective, adverb, noun and verb words have been extracted as such words depict diverse semantic information. After such phrases are extracted the PMI algorithm executes a Latent Semantic Analysis on these phrases to determine their semantic orientation. During the initial phases of sentiment analysis research people generally believed in syntactic influence on the semantic orientation of words. To investigate these relationships in real corpora they generally started with hand-crafted lexicons. Turney used only 1336 hand-labeled adjectives as the seed words.

Most of the semantic orientation detection tasks started with binary classifications, e.g., positive/negative, thums up/thums down, pro/con, like/dislike etc. But gradually a group of independent researchers started thinking about more fine-grained classifications for semantic orientations. They named it emotion analysis or affect computing with the wide perception that the future of human-computer interaction lies in themes such as entertainment, emotions, aesthetic pleasure, motivation, attention, engagement, etc. One of the most important research endeavors in this genre is by (Valitutt et al., 2004). The authors developed a preliminary version of a lexical knowledge base containing words in an affective lexicon connected with a set of affective concepts. This resource (named WORDNET-AFFECT) was developed starting from the lexical knowledge base WORDNET, through a selection and labeling of the affective concepts (represented by sets of synonyms). WORDNET-AFFECT was then

extended taking into account Open Mind, a database of common sense sentences, in which there is a considerable amount of common sense knowledge (Singh, 2002). WORDNET-AFFECT

is also a prior polarity lexicon resource but the semantic classes used here are n-nary such as anger, doubt, competitive, skepticism and pleasure etc.

In the year of 2006, Esuli and Sebastiani (Esuli and Sebastiani, 2006) introduced the idea of SentiWordNet4 which became the most widely used lexical resource for sentiment analysis in the successive years. It is a semi-automatically developed lexical resource, which holds WordNet synsets and prior polarity scores as positivity and negativity. The total occurrence of a particular word in a domain corpus is counted as well as its positive and negative occurrences. Let us consider that the total occurrence of the word "long" in a domain corpus is n and the positive and negative occurrences of the word are p S and n S respectively. Therefore in the developed sentiment lexicon the assigned positivity and negativity scores of the word will be calculated as follows:

$$\text{Positivity} = \frac{S_p}{n}$$

$$\text{Negativity} = \frac{S_n}{n}$$

Four years later, in 2010, the authors released the next version of the resource called SentiWordNet 3.0. (Mihalcea et. al., 2007) have proposed a nice architecture for the development of subjectivity lexicon for resource scarce Romanian language. They started with a small set of seed words for four POS categories: noun, verb, adverb and adjective. The initial seed word list is incremented with an online dictionary along with a small set of manually annotated corpora in a bootstrapping manner. Subjectivity lexicon (Wilson et. al., 2005) is one of the widely used English sentiment lexicon mainly developed from news corpora. The authors showed that lexico-syntactic patterns such as:

X-Drive

Y-got-Angry

help to identify subjective expressions across domains. A subjectivity classifier has been trained on a manually annotated data set and has been used to annotate more data. The data is then used to train the system again by bootstrapping method. (Denecke, 2009) provides an interesting study with the prior polarity scores from the SentiWordNet and shows how these scores could be useful for multiple domains. Two methodologies, one rule-based and another machine learning based, have been proposed in the work. The positivity, negativity and the objectivity scores have been used from the SentiWordNet. A noticeable accuracy has been achieved with the machine learning approach. (Ohana and Tierney, 2009) have reported their

experimentation on review classification using SentiWordNet, which proves the credibility and acceptability of this kind of lexicon resources. A method has been proposed for applying SentiWordNet to derive a data set of document metrics and other relevant features. Experiments have been performed on sentiment classification of film reviews using the SentiWordNet polarity data set. Besides the semantic orientation detection techniques, a number of researchers have attempted for sentiment strength detection. (Thelwall et. al., 2010) have proposed methods for the sentiment strength detection from short informal text. In addition to the research effort concerning the strength detection for multiple emotions (Strapparava and Mihalcea, 2008), there are some works on positive-negative sentiment strength detection. One previous study has used modified sentiment analysis techniques to predict the strength of human ratings on a scale of 1 to 5 for movie reviews (Pang & Lee, 2005). This is a kind of sentiment strength evaluation with a combined scale for positive and negative sentiment. Sentiment strength classification has also been developed for a three level scheme (low, medium, and high or extreme) for subjective sentences or clauses in newswire texts using a linguistic analysis technique that converts sentences into dependency trees reflecting their structure (Wilson et al., 2006). Sentiment analysis researchers have established that prior polarity lexicons are necessary for polarity classification task. Therefore, prior polarity lexicon development endeavor have been noticed for other languages as well, e.g., Chinese (He et. al., 2010), Japanese (Torii et. al., 2010) and Thai (Haruechaiyasaket. al., 2010).

6.2 : **Different Classification Strategies**

It has been reported by several researchers that higher accuracy for prior polarity identification is very hard to achieve. Prior polarity values are approximates. Researchers have argued that prior polarity method along with NLP or other techniques are required for contextual polarity disambiguation. The use of NLP methods or machine learning techniques over the human developed prior polarity lexicon was first pioneered by (Pang et al., 2002). The authors have considered the problem of classifying documents not by topic but by overall sentiment, e.g., determining whether a review is thumbs up (positive) or thumbs down (negative). Using movie reviews data, it has been observed that standard machine learning techniques definitively outperform the human developed prior polarity baseline. However, the three machine learning methods they employed (Naive Bayes, Maximum Entropy classification, and Support Vector Machines) do not perform as well on sentiment classification compared to their performance on traditional topic based categorization. Thereafter a numbers of research attempts like (Salvetti et. al., 2004) have been identified that follow the same system architecture for various other languages and domains.

Another important research attempt to overcome the limitations of the manually augmented prior polarity lexicon is found in (Liu et al., 2003) but the problem domain differs from that of (Pang et al., 2002). Several methods have been presented for assessing the affective qualities,

i.e., emotion classes of natural language and a scenario for its use. Sentiment analysis is a binary classification task whereas the affect sensing is a multi-class problem. A new approach has been demonstrated the use of large-scale real-world knowledge about the inherent affective nature of everyday situations (such as “getting into a car accident”) to classify sentences into “basic” emotion categories. Open Mind[36] Commonsense knowledge has been used as a real world corpus of 400,000 facts about the everyday world. Four linguistic models (Statistical-Syntactic) are combined for robustness as a society of commonsense-based affect recognition. The results suggest that the approach is robust enough to enable plausible affective text user interfaces for future use. This work is also very important in another aspect as it shows the possibility that contextual polarity could be inferred by the syntactic formulations and a formidable accuracy could be reached by this method. The syntactic-statistical techniques for the polarity classification problem have been attempted in several works with good accuracy (Seeker et al., 2009; Moilanen et al., 2010).

Sentiments of people are important because people’s sentiment has great influence on our society. But knowing only the positive or negative aspect of sentiments is not enough because the end users of the proposed Sentiment Analysis systems might look for the comparative or evaluative study for making their own decisions. For example, we always look for a feature wise comparative study before buying any product (Is the product X better than product Y?) or before casting our vote for any candidate (Is Mr. X better than Mr. Y?). To meet such real life necessities, (Liu et. al., 2005) developed a system called Opinion Observer that can analyze and compare opinions available on the Web. The system is such that with a single glance of its visualization, the user is able to clearly see the strengths and weaknesses of the various features of each product in the minds of consumers. A technique based on language pattern mining has been proposed to extract Pros (positive) and Cons (negative) product features in a particular type of reviews. Experimental results show that the technique is highly effective and it outperforms existing methods significantly. Following the same line of hypothesis as (Liu et. al., 2005), (Pang and Lee, 2005) have proposed a sentiment rating technique by viewing the number of stars provided by each customer to each product from customer feedback. The sentiment rating task has been described as a multi-class problem. The standard machine learning technique, Support Vector Machine (SVM) has been used in this setup. The above mentioned sentiment categorization tasks make an implicit assumption that a single score can express the polarity of an opinion text. However, multiple opinions on related matters are often intertwined throughout a text. For example, a restaurant review may express judgment on food quality as well as the service and ambience of the restaurant. Rather than accumulating these aspects into a single score, people may get interested to know the aspectual sentiment separately. Therefore to provide such facility, (Snyder and Barzilay, 2007) have proposed their Multiple Aspect Ranking technique using the Good Grief Algorithm. The Good Grief algorithm guides the prediction of individual rankers by analyzing meta-relations between opinions, such as agreement and contrast. Probably this is the first attempt when researchers started using data mining based semantic association models for polarity classification task. This kind of

modeling has been attempted by other researchers later (Speriosu et. al., 2011). The Text Retrieval Conference (TREC) Polarity Classification of Blog track[37] 2008 brought together researchers to share their knowledge and compare the efficiency of their proposed techniques. Most of the submitted runs for the task used a two-stage approach (prior polarity identification followed by NLP or other techniques). Only 12 runs out of the submitted 191 runs did not adopt this strategy. The three opinion-finding approaches out of these 12 runs that were consistently effective across the entire provided baseline have been focused in the present work. The approach by University of Illinois at Chicago (Jia et al., 2008) achieved the best average improvement over the standard topic-relevance baselines (an average of 11.76% improvement) for the opinion-finding. Sentence level and document level polarity classification models have been developed and finally the polarity scores have been accumulated to generate the final result. The sentence level classifier is a simple SVM based classifier that classifies a query relevant opinion sentence as either positive or negative. Two approaches were proposed at document level, a Heuristic Rule Based Model and the Decision Tree Model. The Heuristic Rule Based polarity classification system was developed based on the following intuition: a document is positive (negative) if it only contains positive (negative) relevant opinions. If the document contains both kinds of opinions, it needs further analysis. If the positive (negative) relevant opinions are significantly stronger than the negative (positive) relevant opinions, the opinion polarity of this document should be positive (negative). The Decision Tree Model is a machine learning method that improves the document-level opinion polarity classification accuracy. A vector of polarized words/phrases is formed for each document whose polarity is determined initially by the sentence level classifier. This research effort is very significant because it shows the clear distinction between the sentence level and the document level polarity classification. Later, many other researchers (Somsundaram and Wiebe, 2009) have considered the polarity classification problems distinctly at sentence level and document level. The approach by (Lee et. al., 2008) has used a domain specific lexicon based approach. In addition to SentiWordNet, the authors have used Amazon's product review corpus and product specification corpus to create the opinionated lexical resource. This clearly shows that domain knowledge is required for polarity classification along with generic prior polarity lexicons like SentiWordNet. The accuracy of a polarity classifier mainly depends on the handling of unknown words or new words. The same conclusion has been drawn by several other researchers later (Aue and Gamon, 2009; Takamura et. al., 2005) as well. (He et. al., 2008) have used their domain specific divergence model for polarity classification task. The work is based on the hypothesis that the semantic orientation of prior polarity lexicon from a preprocessed dictionary may vary in the current domain. The authors have enhanced a dictionary-based approach by automatically building an internal opinion dictionary from the provided corpus collection itself. This approach measures the opinionated discrimination property of each term in the dictionary using information theoretic divergence measure based on the relevance assessments at context level.

6.3 : Human Psychology to Solve the Sentiment Analysis

The Sentiment Analysis research has become quite matured after a few decades of research. As a result, a few systems like Twitter Sentiment Analysis Tool (http://twittersentiment.appspot.com/), TweetFeel (http://www.tweetfeel.com/) are available in the World Wide Web since last few years. More research efforts are necessary to meet the satisfaction level of the end users (Liu, 2010). The main issue is that there are many conceptual rules that govern sentiment and there are even more clues (possibly unlimited) that can convey these concepts from realization to verbalization of a human being. Human psychology may provide the unrevealed clues and govern the sentiment realization. Human psychology relates to social, cultural, behavioral and environmental aspects of civilization. The important issues that need attention include how various psychological phenomena can be explained in computational terms and the identification of the various Artificial Intelligence (AI) concepts and computer modeling methodologies that are most useful from the psychologist's point of view. An important research endeavor could be noticed supporting this notion in the form of a workshop "Sentiment Analysis where AI meets Psychology (SAAIP 2011)[38]" held as part of the International Joint Conference on NLP (IJCNLP 2011).

(Cambria et al., 2011) did a wonderful contribution in this direction. They introduced a new paradigm, called Sentic Computing[39], in which an emotion representation and a Common Sense[40] (Cambria et al., 2009) based approach have been used to infer affective states from short texts over the web. The innovation of Sentic Computing is a result of in-depth scientific cultivation by several other researchers over two decades. Some of those important research attempts which made the avenue to the present Sentic computing are reported below.

The term 'sentic' is derived from the Latin word 'sentire', the root of words like sentiment and sensation. It was first adopted in 1977 by Manfred Clynes (Clynes, 1977), who discovered that when people have emotional experience, their nervous system always responds in a characteristic way which is measurable. Sentic Computing is part of the efforts in the fields of computer science, psychology, linguistics, sociology and cognitive science, to develop a kind of computing that relates to or arises from or influences emotions (Picard, 1997). The approach adopted by (Liu et. al., 2003) exploits a Common Sense knowledge base to extract affective information from emails using the standard notion of basic emotions provided by Ekman[41]. Nowadays, researchers use a much richer semantic network, ConceptNet[42] (Havasi et. al., 2007), with almost 10,000 concepts and a set of 72,000+ features extracted from the Open Mind corpus along with the power of cumulative analogy provided by AnalogySpace, a process which reveals large-scale patterns in the data, smoothes over noise and predicts new knowledge.

The aim in Sentic Computing is to develop emotion-sensitive systems that can measure how much:

1. The user is happy with the service provided?

2. The user is interested in the information supplied?

3. The user is comfortable with the interface?

4. The user is keen on using the application?

Thus, in Sentic Computing the user's affective states are organized around four independent dimensions: Pleasantness, Attention, Sensitivity and Aptitude. This model is a variant of Plutchik's wheel of emotions (Plutchik, 2001) and constitutes an attempt to emulate Marvin Minsky's conception of emotions (The Emotion Machine13: Minsky, 2006). Minsky sees the mind as a collection of thousands of different resources and believes that our emotional states result from turning some set of these resources on and turning another set of them off. Each such selection identifies how we think by changing our brain's activities: the state of anger, for example, appears to select a set of resources that help us to react with more speed and strength while also suppressing some other resources that usually make us act prudently.

6.4: Resource Acquisition

In the present work, the polarity classification experiments have been carried out for Bengali language. The Sentiment Analysis task for a new language demands linguistic resources like gold standard annotated data and other NLP tools. The basic polarity classification task in the present work started with syntactic dependency (Liu et. al., 2003). Therefore, a dependency parser is necessary for the experiments. Bengali is a resource scarce language and no Bengali Dependency parser was available when the work started. Thus, the development of Bengali Dependency parser was identified an important task.

6.5: Corpus

All the experiments in the present work have been carried out on Bengali news corpus. News text can be divided into two main types:

(1) News reports that aim to objectively present factual information and

(2) Opinionated articles that clearly present authors' and readers' views, evaluation or judgment about some specific events or persons.

Type (1) is supposed to be the common practice in newspapers, and

Type (2) appears in sections such as 'Editorial', 'Forum' and 'Letters to the editor'. 'Reader's opinion' section or 'Letters to the Editor' Section from the web archive of a popular Bengali newspaper have been identified as the relevant corpus in Bengali. A brief statistics about the corpus have been reported

in the Table The corpus is then manually annotated. The annotation scheme used in the corpus annotation is reported in Figure The positive algebraic sign in the feature structure ("<fs af=,+,")

depicts the phrase polarity as positive and the negative algebraic sign in the feature structure (“<fs af=,_,”) depicts the phrase polarity as negative .

Corpus Statistics	
Total number of documents in the corpus	20
Total number of sentences in the corpus	447
Avgerage number of sentences in a document	22
Total number of wordforms in the corpus	5761
Avgerage number of wordforms in a document	288
Total number of distinct wordforms in the corpus	3435

Table 7: Statistics on Bengali Polarity Annotated News Corpus

2	((	CCP	
2.1	যেমন	CC	
3	((	NP	<fs af=',+,,,,,,' name='?'>
3.1	মঙ্গলজনক		NN

Figure 1: Bengali Corpus Polarity Annotation Scheme

6.6 : The Syntactic Polarity Classifier

The two step methodology, i.e., use of prior polarity lexicon followed by any NLP technique is the standard method for the polarity classification task. For the NLP technique, the Syntactic-Statistical classification NLP technique has been used (Das and Bandyopadhyay, 2010(a));(Das and Bandyopadhyay, 2010(h)). The syntactic clue directly helps to understand the relation between the localized semantic orientation, i.e., word level semantic orientation and the contextual semantic orientation, i.e., word/phrase/sentence level semantic orientation. In the following example sentence, the localized semantic orientation at word level, ভালো (good) could be obtained directly from the prior polarity lexicon as positive.

He is not a good+ boy.

সে ভালো+ ছেলে নয়

The negation word 'not' changes the contextual semantics in the opposite direction, i.e., negative. To understand this contextual feature, the syntactic relationship helps as the word "not (নয়)" has a modifier relationship with the word "good (ভালো)" (modified). Therefore, it is very easy to infer the resultant contextual semantic orientation of the sentence as negative.

Moreover the syntax sometime helps to predict the semantic orientation of any new word. Let us take a look at the following example sentence.

This is ugly- and smelly.

এটি বিশ্রী- এবং কটগন্ধযুক্ত

Let us consider that the prior polarity lexicon only covers "ugly (বিশ্রী)" and not the "smelly (কটগন্ধযুক্ত)". As the semantic orientation of the word "ugly (বিশ্রী)" is negative it is more or less obvious that the semantic orientation of the new word "smelly (কটগন্ধযুক্ত)" will be the same because it has been observed that generally words with same orientation are syntactically joined with "and" and words with orthogonal semantic orientation are syntactically joined with "but/rather/either…etc" as seen in the following example sentence.

Good+ but costly-

ভালো+ কিন্তু দামী-

Several other researchers (Liu et. al., 2003; Seeker et. al., 2009; Moilanen et. al., 2010) have also identified the same linguistic phenomena. But there are exceptions like, "The Good Bad and Ugly". In the famous movie title, "Bad and Ugly" is syntactically joined by the conjunct "and" with the word "Good". The three adjectives in the title metaphorically refer to three entities or three persons who are the characters in the movie. Such exceptions are rare in the language.

It has also been observed that localized syntax helps to understand the discourse level sentimental semantics to some extent (Somsundaram, 2009). For example, the following sentences are from two different paragraphs from the same document.

The reason behind the electoral disaster is the wrong **policy of the previous Government.**

পূর্বতন সরকারের ভুল নীতি ভোটে ভরাডুবির অন্যতম কারন

We will not follow the **strategy of the previous government**

আমরা পূর্বতন সরকারের নীতি অনুসরণ করব না

In the first sentence the word “wrong (ভুল)” is modifying the phrase “strategy of the previous Government (পূর্বতন সরকারের নীতি) and it is negative. Therefore in the same scope of the document it is very likely that a single author will not sentimentally differ too much regarding the same topic and thus the final semantic orientation of the second sentence is likely to be positive as it includes a negation. But it is very hard to assimilate this kind of knowledge into the Syntactic-Statistical polarity classifier. An in-depth semantic tagging at the discourse level is required for this kind of work.

6.7: Features Extraction

The standard machine learning method Support Vector Machine (SVM)[44] can be used for the syntactic statistical polarity classifier. The SVM has a few advantages over the other existing machine learning techniques that depend on the data being analyzed. The typical scenario for the SVM is when the data are not regularly distributed or have an unknown distribution. Sentiment analysis data is a perfect example of this type. No one can predict in which order the positive or negative words will occur in a text, i.e., there is no regular distribution. It completely depends on the psychological forces of the situation that were in effect when the document was written by a particular writer. Moreover sentiment is not a linguistic phenomenon and it is nearly impossible to identify the concrete set of psychological or cognitive features from the written text. A detailed psycho-linguistic study is necessary which demands more and more reliable linguistic tools but unfortunately such tools are unavailable for Bengali language. SVM works well with less numbers of distinct informative features which is essential for working with a new language. SVM provides a good out-of-sample generalization. It means that, by choosing an appropriate generalization grade, SVMs can be robust, even when the training sample has some bias or limitations (Auria and Moro, 2008). To support the argumentation in favor of SVM, experiments were conducted using the CRF machine learning technique with the same data and setup.

SVM treats opinion polarity identification as a sequence tagging and pattern-matching task, acquiring symbolic patterns that rely on both the syntax and lexical semantics of a phrase and sentence. Several word level features are extracted using different tools from the input sentences. The feature identification starts with Part Of Speech (POS) categories and the exploration is continued with other features like chunk, functional word, SentiWordNet (Bengali), stemming cluster, Negative word list and Dependency tree features. The feature extraction for any Machine Learning task is crucial since proper identification of the entire features directly affects the performance of the system. Functional word, SentiWordNet (Bengali) and Negative word list features are fully dictionary based. On the other hand, POS, chunk, stemming cluster and dependency tree features are extractive.

6.8: Part Of Speech (POS)

It has been shown (Hatzivassiloglou et. al., 2000; Chesley et. al., 2006) that opinion bearing words in sentences are mainly adjective, adverb, noun and verbs. Many opinion-topic identification systems, like (Nasukawa et. al., 2003) are based on adjective or adverb words. The Bengali Shallow Parser[45] developed under the "Indian Languages to Indian Languages machine Translation (IL-ILMT)" project funded by Department of Information Technology; Government of India. The shallow parser gives the analysis of a sentence in terms of morphological analysis, POS tagging, Chunking, etc. Apart from the final output, intermediate output of individual modules is also available. All outputs are in Shakti Standard Format (SSF).

6.9 : Chunk

In the Syntactic-Statistical polarity classifier local dependencies like chunk boundaries and chunk member information are very important features. It is not unusual for two annotators to identify the same expression as a polar element in the text, but they could differ in how they mark the boundaries, such as the difference between 'such a disadvantageous situation' and 'such…disadvantageous' (Wilson and Wiebe, 2003). Similar fuzziness appeared in the marking of polar elements in the present task, such as 'কেন্দ্রীয় দলের দূর্নীতিতে' (corruption of central team) and 'দূর্নীতিতে' (corruption). Hence the hypothesis is to stick to the automatically assigned chunk labels only to avoid any further ambiguity. Chunk level information is effectively used as a feature in the supervised classifier. Chunk labels are defined as B-X (Beginning), I-X (Intermediate) and E-X (End), where X is the chunk label.

6.10 : Functional word

Function words in a language are high frequency words and these words generally do not carry any opinionated information. But function words help many times to understand the syntactic pattern of a sentimental text. A list of 253 functional words is collected from the Bengali corpus. First a unique high frequency word list is generated where the assumed threshold frequency is considered as 20. Then the list is manually corrected. The function word feature is very important to disambiguate the contextual polarity for unknown words.

Prior Polarity Lexicon

The classical two step methodology for the polarity classification problem, i.e., prior polarity lexicon followed by NLP techniques for further contextual polarity disambiguation, has been followed in the present work. The developed Bengali SentiWordNet[43] is used as the prior polarity lexicon in the present work. Words that are present in the SentiWordNet carry sentiment information. The prior polarity lexicon features are individual sentiment words or word n-grams (multiword entities) with polarity values either positive or negative. Positive and negative polarity measures are treated as a binary feature in the supervised classifier. Words

which are collected directly from the SentiWordNet are tagged with positivity or negativity scores.

6.11 : Stemming cluster

Several words in a sentence that carry opinion information may be present in inflected forms. Stemming is necessary for such inflected words before they can be searched in the appropriate lists. Due to non-availability of good stemmers in Indian languages, especially in Bengali, a stemmer based on stemming cluster technique has been evolved. This stemmer analyzes prefixes and suffixes of all the word forms present in a particular document. Words that are identified to have the same root form are grouped in a finite number of clusters with the identified root word as the cluster center. Details can be found in (Das and Bandyopadhyay, 2010(l)).

6.12 : Negative words

Negative words like no (না), not (নয়) etc. does not carry any sentiment information but these words relationally affect the resultant polarity of any polar phrase. A manually edited list of negative words has been used as a binary feature in the SVM classifier.

6.13 : Dependency Tree feature

Dependency relations are the most crucial feature in the Syntactic-Statistical polarity classifier. The feature extractor module searches the dependency tree using the breadth-first search technique to identify syntactically related nodes and their mutual relations. The purpose of the feature is to encode dependency structure between related polar phrases.

6.14 : Performance of the Syntactic Polarity Classifier

The evaluation result of the SVM-based polarity classification task for Bengali is presented in Table 8. The evaluation results of the system for the positive and negative polarity classes are mentioned separately in Table.

Language	Precision	Recall
Bengali	70.04%	63.02%

Table 8: The Overall Performance of Polarity Classification for Bengali

Polarity	Precision	Recall
Positive	56.59%	52.89%
Negative	75.57%	65.87%

Table 9: Polarity Wise Performance of Polarity Classification for Bengali

To understand the effects of various features on the performance of the system the feature ablation method has been studied. The dictionary based approach using only the SentiWordNet has accuracy (precision) of 47.60% and this may be considered as the baseline system. It may be observed from the Table 3 that incremental use of other features like negative word, functional word, parts of speech, chunk and tools like stemming cluster has improved the precision of the system to 66.8%. Thus an increase of 19.2% in precision over the baseline system has been obtained. Further use of syntactic feature in terms of dependency relations has improved the system precision to 70.04%. Thus an increase of 3.6% in precision has been obtained due to the use of syntactic feature. The feature ablation method proves the effectiveness of the two step polarity classification technique. The prior polarity lexicon, i.e., completely dictionary based approach produces 47.60% precision and further improvement of the system could be achieved using various NLP techniques.

To support the arguments for choosing SVM machine learning method, the same classification problem was attempted using CRF machine learning technique with the same data and setup. The resulting accuracy of the CRF based model with precision 61.23% and recall 55.0% is much less than the SVM based model. The same feature ablation method as reported in the Table 3 was applied on the CRF based model. It has been noticed that the accuracy level is more or less same till the dictionary features and lexical features (SentiWordNet + Negative Word + Stemming Cluster + Functional Word + Parts Of Speech) are used. But it is hard to increase the performance level of the CRF based model when the syntactic features like chunk and dependency relations are used. SVM machine learning technique works excellent to normalize this dynamic situation.

Features	Performance
SentiWordNet	47.60%
SentiWordNet + Negative Word	50.40%
SentiWordNet + Negative Word + Stemming Cluster	56.02%
SentiWordNet + Negative Word + Stemming Cluster + Functional Word	58.23%
SentiWordNet + Negative Word + Stemming Cluster + Functional Word + Parts Of Speech	61.90%
SentiWordNet + Negative Word + Stemming Cluster + Functional Word + Parts Of Speech +Chunk	66.80%
SentiWordNet + Negative Word + Stemming Cluster + Functional Word + Parts Of Speech + Chunk +Dependency tree feature	70.04%

Table 10: Performance of the Syntactic Polarity Classifier by Feature Ablation

7. Sentiment Lexicon Generation

By now, it should be quite clear that words and phrases that convey positive or negative sentiments are instrumental for sentiment analysis. This chapter discusses how to compile such words lists. In the research literature, sentiment words are also called ***opinion words***, ***polar words***, or ***opinion bearing words***. Positive sentiment words are used to express some desired states or qualities while negative sentiment words are used to express some undesired states or qualities. Examples of positive sentiment words are সুন্দর,দারুন, and অসাধারন. Examples of negative sentiment words are খারাপ, ভয়াবহ, and অসহায়. Apart from individual words, there are also sentiment phrases and idioms, e.g.,কারো পৌষ মাস, কারো সর্বনাশ. Collectively, they are called sentiment lexicon (or opinion lexicon). For easy presentation, from now on when we say sentiment words, we mean both individual words and phrases.

Sentiment words can be divided into two types, *base type* and *comparative type*. All the example words above are of the base type. Sentiment words of the comparative type (which include the superlative type) are used to express comparative and superlative opinions. Examples of such words are better, *worse, best, worst*, etc., which are comparative and superlative forms of their base adjectives or adverbs, e.g., *good* and *bad*. Unlike sentiment words of the base type, sentiment words of the comparative type do not express a regular opinion on an entity but a comparative opinion on more than one entity, e.g., "*Pepsi tastes better than Coke*." This sentence does not express an opinion saying that any of the two drinks is good or bad. It just says that compared to Coke, Pepsi tastes better.

Researchers have proposed many approaches to compile sentiment words. Three main approaches are: *manual approach*, *dictionary-based approach*, and *corpus-based approach*. The manual approach is labor intensive and time consuming, and is thus not usually used alone but combined with automated approaches as the final check, because automated methods make mistakes. Below, we discuss the two automated approaches. Along with them, we will also discuss the issue of factual statements implying opinions, which has largely been overlooked by the research community.

7.1: Dictionary-based Approach

Using a dictionary to compile sentiment words is an obvious approach because most dictionaries (e.g., WordNet (Miller et al., 1990)) list synonyms and antonyms for each word. Thus, a simple technique in this approach is to use a few seed sentiment words to bootstrap based on the synonym and antonym structure of a dictionary. Specifically, this method works as follows: A small set of sentiment words (seeds) with known positive or negative orientations is first collected manually, which is very easy. The algorithm then grows this set by searching in the WordNet or another online dictionary for their synonyms and antonyms. The newly found words are added to the seed list. The next iteration begins. The iterative process ends when no more new words can be found. This approach was used in (Hu and Liu, 2004). After the process

completes, a manual inspection step was used to clean up the list. A similar method was also used by Valitutti, Strapparava and Stock (2004). Kim and Hovy (2004) tried to clean up the resulting words (to remove errors) and to assign a sentiment strength to each word using a probabilistic method. Mohammad, Dunne and Dorr (2009) additionally exploited many antonym-generating affix patterns like X and disX (e.g., honest–dishonest) to increase the coverage.

A more sophisticated approach was proposed in (Kamps et al., 2004), which used a WordNet distance based method to determine the sentiment orientation of a given adjective. The distance d(t1, t2) between terms t1 and t2 is the length of the shortest path that connects t1 and t2 in WordNet. The orientation of an adjective term t is determined by its relative distance from two reference (or seed) terms good and bad, i.e., SO(t) = (d(t, bad) − d(t, good))/d(good, bad). t is positive iff SO(t) > 0, and is negative otherwise. The absolute value of SO(t) gives the strength of the sentiment. Along a similar line, Williams and Anand (2009) studied the problem of assigning sentiment strength to each word.

In (Blair-Goldensohn et al., 2008), a different bootstrapping method was proposed, which used a positive seed set, a negative seed set, and also a neutral seed set. The approach works based on a directed, weighted semantic graph where neighboring nodes are synonyms or antonyms of words in WordNet and are not part of the seed neutral set. The neutral set is used to stop the propagation of sentiments through neutral words. The edge weights are pre-assigned based on a scaling parameter for different types of edges, i.e., synonym or antonym edges. Each word is then scored (giving a sentiment value) using a modified version of the label propagation algorithm in (Zhu and Ghahramani, 2002). At the beginning, each positive seed word is given the score of +1, each negative seed is given the score of -1, and all other words are given the score of 0. The scores are revised during the propagation process. When the propagation stops after a number of iterations, the final scores after a logarithmic scaling are assigned to words as their degrees of being positive or negative.

In (Rao and Ravichandran, 2009), three graph-based semi-supervised learning methods were tried to separate positive and negative words given a positive seed set, a negative seed set, and a synonym graph extracted from the WordNet. The three algorithms were Mincut (Blum and Chawla, 2001), Randomized Mincut (Blum et al., 2004), and label propagation (Zhu and Ghahramani, 2002). It was shown that Mincut and Randomized Mincut produced better F scores, but label propagation gave significantly higher precisions with low recalls.

Hassan and Radev (2010) presented a Markov random walk model over a word relatedness graph to produce a sentiment estimate for a given word. It first uses WordNet synonyms and hypernyms to build a word relatedness graph. A measure, called the mean hitting time h(i|S), was then defined and used to gauge the distance from a node i to a set of nodes (words) S, which is the average number of steps that a random walker, starting in state i ∉ S, will take to enter a state k ∈ S for the first time. Given a set of positive seed words S+ and a set of negative seed words S−, to estimate the sentiment orientation of a given word w, it computes the hitting

times h(w|S+) and h(w|S−). If h(w|S+) is greater than h(w|S−), the word is classified as negative, otherwise positive. In (Hassan et al., 2011), this method was applied to find sentiment orientations of foreign words. For this purpose, a multilingual word graph was created with both English words and foreign words. Words in different languages are connected based on their meanings in dictionaries. Other methods based on graphs include those in (Takamura, Inui and Okumura, 2005) and (Takamura, Inui and Okumura, 2007; Takamura, Inui and Okumura, 2006).

Esuli and Sebastiani (2005) used supervised learning to classify words into positive and negative classes. Given a set P of positive seed words and a set N of negative seed words, the two seed sets are first expanded using synonym and antonym relations in an online dictionary (e.g., WordNet) to generate the expanded sets P' and N', which form the training set. The algorithm then uses all the glosses in the dictionary for each term in P' U N' to generate a feature vector. A binary classifier is then built using different learning algorithms. The process can also be run iteratively. That is, the newly identified positive and negative terms and their synonyms and antonyms are added to the training set, an updated classifier can be constructed and so on. In (Esuli and Sebastiani, 2006), the authors also included the category objective. To expand the objective seed set, hyponyms were used in addition to synonyms and antonyms. They then tried different strategies to do the three-class classification. In (Esuli and Sebastiani, 2006), a committee of classifiers based on the above method was utilized to build the SentiWordNet, a lexical resource in which each synset of WordNet is associated with three numerical scores Obj(s), Pos(s) and Neg(s), describing how Objective, Positive, and Negative the terms contained in the synset are. The method of Kim and Hovy (2006) also started with three seed sets of positive, negative, and neutral words. It then finds their synonyms in WordNet. The expanded sets, however, have many errors. The method then uses a Bayesian formula to compute the closeness of each word to each category (positive, negative, and neutral) to determine the most probable class for the word.

7.2: Corpus-based Approach

The corpus-based approach has been applied to two main scenarios:

(1) given a seed list of known (often general-purpose) sentiment words, discover other sentiment words and their orientations from a domain corpus, and
(2) Adapt a general-purpose sentiment lexicon to a new one using a domain corpus for sentiment analysis applications in the domain.

However, the issue is more complicated than just building a domain specific sentiment lexicon because in the same domain the same word can be positive in one context but negative in another. Below, we discuss some of the existing works that tried to deal with these problems. Note that although the corpus-based approach may also be used to build a general-purpose sentiment lexicon if a very large and very diverse corpus is available, the

dictionary-based approach is usually more effective for that because a dictionary has all words.

One of the key and also early ideas was proposed by Hazivassiloglou and McKeown (1997). The authors used a corpus and some seed adjective sentiment words to find additional sentiment adjectives in the corpus. Their technique exploited a set of linguistic rules or conventions on connectives to identify more adjective sentiment words and their orientations from the corpus. One of the rules is about the conjunction AND, which says that conjoined adjectives usually have the same orientation. For example, in the sentence, "*This car is beautiful and spacious*," if "*beautiful*" is known to be positive, it can be inferred that "*spacious*" is also positive. This is so because people usually express the same sentiment on both sides of a conjunction. The following sentence is not likely, "*This car is beautiful and difficult to drive.*" It is more acceptable if it is changed to "*This car is beautiful but difficult to drive.*" Rules were also designed for other connectives, i.e.,

OR, BUT, EITHER–OR, and NEITHER–NOR.

This idea is called sentiment consistency. In practice, it is not always consistent. Thus, a learning step was also applied to determine if two conjoined adjectives have the same or different orientations. First, a graph was formed with same- and different orientation links between adjectives. Clustering was then performed on the graph to produce two sets of words: positive and negative.

Kanayama and Nasukawa (2006) extended the approach by introducing the concepts of intra-sentential (within a sentence) and inter-sentential (between neighboring sentences) sentiment consistency, which they call coherency. The intra-sentential consistency is similar to the idea above. Inter-sentential consistency simply applies the idea to neighboring sentences. That is, the same sentiment orientation is usually expressed in consecutive sentences. Sentiment changes are indicated by adversative expressions such as but and however. Some criteria were also proposed to determine whether to add a word to the positive or negative lexicon. This study was based on Japanese text and was used to find domain dependent sentiment words and their orientations. Other related work includes those in (Kaji and Kitsuregawa, 2006; Kaji and Kitsuregawa, 2007).

7.3: Desirable and Undesirable Facts

Sentiment words and expressions that we have discussed so far are mainly subjective words and expressions that indicate positive or negative opinions. However, as mentioned earlier, many objective words and expressions can imply opinions too in certain domains or contexts because they can represent desirable or undesirable facts in these domains or contexts.

In (Zhang and Liu, 2011b), a method was proposed to identify nouns and noun phrases that are aspects and also imply sentiments in a particular domain. These nouns and noun

phrases alone indicate no sentiments, but in the domain context they may represent desirable or undesirable facts. For example, "*valley*" and "*mountain*" do not have any sentiment connotation in general, i.e., they are objective. However, in the domain of mattress reviews, they often imply negative opinions as in "*Within a month, a valley has formed in the middle of the mattress*." Here, "*valley*" implies a negative sentiment on the aspect of mattress quality. Identifying the sentiment orientations of such aspects is very challenging but critical for effective sentiment analysis in these domains.

The algorithm in (Zhang and Liu, 2011b) was based on the following idea:

Although such sentences are usually objective with no explicit sentiments, in some cases the authors/reviewers may also give explicit sentiments, e.g., "*Within a month, a valley has formed in the middle of the mattress, which is terrible*." The context of this sentence indicates that "valley" may not be desirable. Note that this work assumed that the set of aspects which are nouns and noun phrases are given. However, the problem with this approach is that those aspects (nouns and noun phrases) with no implied sentiment may also be in some positive or negative sentiment contexts, e.g., "*voice quality*" in "*The voice quality is poor.*" To distinguish these two cases, the following observation was used.

7.3.1 Observation: For normal aspects which themselves don't have positive or negative connotations, people can express different opinions, i.e., both positive and negative. For example, for aspect "*voice quality*", people can say "*good voice quality*" and "*bad voice quality*". However, for aspects which represent desirable or undesirable facts, they often have only a single sentiment, either positive or negative, but not both. For example, it is unlikely that both the following two sentences appear: "*A bad valley has formed*" and "*a good valley has formed".*

With this observation in mind, the approach consists of two steps:

1. *Candidate identification*: This step determines the surrounding sentiment context of each noun aspect. If an aspect occurs in negative (respectively positive) sentiment contexts significantly more frequently than in positive (or negative) sentiment contexts, it is inferred that its polarity is negative (or positive). This step thus produces a list of candidate aspects with positive opinions and a list of candidate aspects with negative opinions.
2. *Pruning*: This step prunes the two lists based on the observation above. The idea is that when a noun aspect is directly modified by both positive and negative sentiment words, it is unlikely to be an opinionated aspect.
   Two types of direct dependency relations were used.
   Type 1: O →O-Dep → F
   It means O depends on F through the relation O-Dep, e.g., "*This TV has a good picture quality.*"
   Type 2: O → O-Dep → H ← F-Dep ← F

It means both O and F depend on H through relations O-Dep and FDep respectively, e.g., "*The springs of the mattress are bad*."
where O is a sentiment word, O-Dep / F-Dep is a dependency relation. F is the noun aspect. H means any word. For the first example, given aspect "*picture quality*", we can identify its modification sentiment word "*good."* For the second example, given aspect "*springs*", we can get its modification sentiment word "*bad*". Here H is the word "*are*".

This work is just the first attempt to tackle the problem. Its accuracy is still not high. Much further research is needed.

7.4: 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. The word list is then further expanded with the corresponding synonyms of the words in the wordlist.

7.4.1 Collecting Bangla words

To make our desire Bangla lexicon, the first step is collecting all Bangla words. Initially 56909 Bangla words are taken from an android Bangle Dictionary. But it is not a complete dictionary, that's why, many words are also taken from others bilingual dictionaries Ovidhan and Samsad. Day by day , this Bangla lexicon is being enriched by manual process and website parsing.

7.4.2 Parts of Speech(POS) Tagging :

We use Online POS tagger tool to tag all Bangla words. In this process, at first we send Bangla words to the online tagger tool and collect all tagged word from the website. For parsing the POS tag from website we use BeautifulSoup a Python package for parsing HTML documents. We use this POS tagger for its accuracy. We have tried many other both online and offline Bangla POS tagger tool before but unfortunately the result was not satisfactory. This online tagging tool covers others language like English, Hindi, Chinese, Dutch etc. It is a strong online tool that is powered by Python NLTK library.

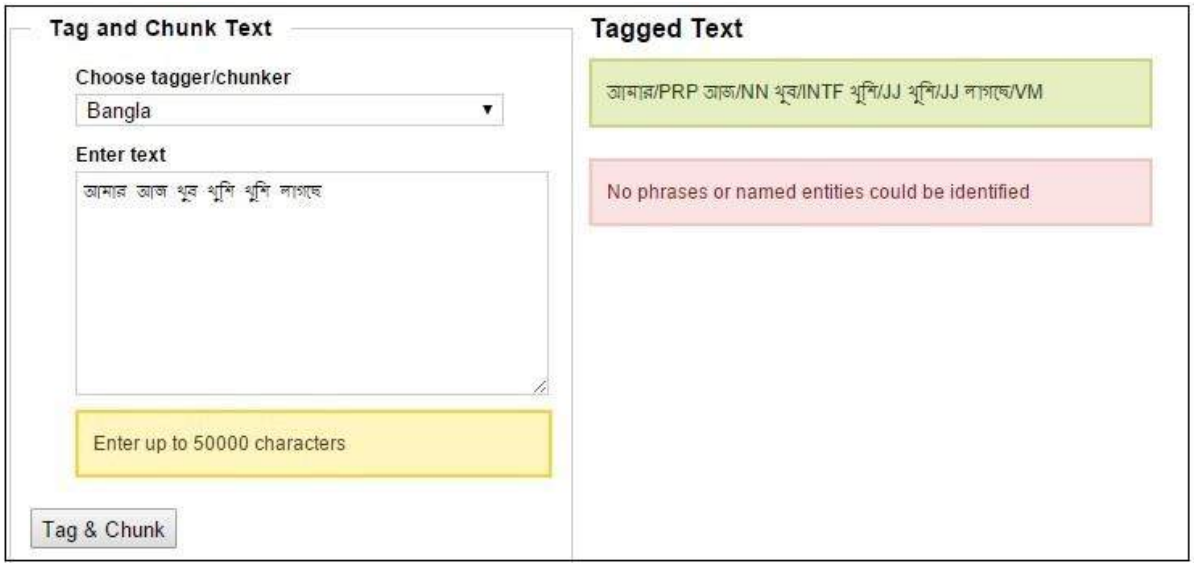


Figure 2: Bangla Parts of Speech (POS) tagging

Here we use an example to show how parts of speech tagging is retrieved. We send Bangla sentence "আমার আজ খুব খুশি খুশি লাগছে" to the online tagger tool using *BeautifulSoap*. *BeautifulSoap* is a powerful python library for web parsing. After tagging all words, we received আমার/PRP ,আজ/NN ,খুশি/JJ ,খুশি/JJ ,লাগছে/VM as an output. Here *PRP* means *Pronoun*, *NN* means *Noun*, *JJ* means *Adjective* and *VM* means *Verb*. We have a complete list of these tags.

Serial	POS Tag	Meaning	No. of Words In Dataset	Examples
1	CC	Conjuncts (Coordinating and Subordinating)	995	অংশটুকু[portion], অথচ[yet],অথবা[or], স্বয়ং[self]
2	DEM	Demonstrative	236	অথই[bottomless], বটেই[forsooth], ভালোই[fine], শুধুই[only]
3	INJ	Interjection	37	আচ্ছা[okay], চিঁ-চিঁ[squeak], ছোঁ[clutch], ঝাঁ-ঝাঁ[glow], সোঁ[smack]
4	INTF	Intensifier	16	অতি[vastly], অত্যন্ত[highly], একটু[a little], বেশ[well], সম্পূর্ণ[complete], সবচেয়ে[most]

5	JJ	Adjective (Modifier of Noun)	8459	অননুতপ্ত[unrepentant], অননুভূত[unfelt], নিমজ্জিত[submerged], ফলিত[applied], হ্রাসপ্রাপ্ত[reduced]
6	NEG	Negative	11	না[no / not / never], নাহ[not]
7	NN	Common Nouns	34734	অনুসন্ধানী[examining], অধিবিদ্যার[metaphysical], উপায়ান্তর[alternative], ঘটনাবহুল[eventful], সূর্যালোক[sunlight]
8	NNC	Compound Noun	2	অধ্যাপক[professor], বিশপ[bishop]
9	NNP	Proper Nouns (name of person)	928	অগ্নিকাণ্ড[fire], আশ্বাস[assure], উপোস[fast], দণ্ড[penalty], শ্বাস-প্রশ্বাস[breathing]
10	NST	Noun Denoting Spatial and Temporal Expressions	36	অদূরে[yonder], আগে[before], উপরে[above], কাছে[near], পিছনে[back], সঙ্গে[with], সামনের[front]
11	PRP	Pronoun	191	আমি [I], তাঁর[his], যিনি[who], সবাই[everyone]
12	PSP	Postposition	32	অনুযায়ী[according], উদ্দেশ্যে[purposes], জন্য [for], দ্বারা[with], মাধ্যমে[through]
13	QC	Cardinals	26	একটি/ একটার [one], অষ্টাদশ [eighteen], হাজার[thousand]
14	QF	Quantifiers	41	অনেক[many], অর্ধেক[half], অল্প[short], কোনো[any], কয়েক[few], সামান্য[little]
15	QO	Ordinal	4	প্রথম [first], দ্বিতীয়[second], তৃতীয়[third], পনেরই[fifteenth]
16	RB	Adverb(Modifier of Verb)	106	আবার[again], আরো[more], একমাত্র[only], একান্তভাবে[exclusively], ক্রমশ[stepwise], সাধারণভাবে[generally]
17	RDP	Reduplications	14	আলাদা[different], কিছু[some], খুঁজে[out], পৃথক[separate], ফিরে[back], যুগ[era]
18	RP	Particles	15	ইত্যাদি/ প্রভৃতি [etc.], করিয়া[by], ভাবে[way], যেন[as though]
19	SYM	Symbol	4	আশ্চর্যবোধক[exclamation], চাটু [pan], ড্যাশ[dash], ।[.]
20	UT	Quotative	2	বলতে[say], বলে[say]
21	VAUX	Verb Auxiliary (Any verb, present besides main verb shall be marked as auxiliary verb)	60	আসছে[coming], উঠছে [becoming], করা[done], দেওয়া[given], যাচ্ছে[going], লওয়া[take]
22	VM	Verb Main (Finite or Non-Finite)	10565	অগ্রধাবন[rush], আরক্তবদন[apoplectic], উৎপাদন[production], রুক্ষভাবে[rudely], হারানো[losing], সন্নিকটে[close], প্রজ্বলন[lighting]

23	WQ	Question Words	10	কি[what], কিসের[whereof], কে[who], কেন[why], কোথায়[where]
24	XC	Compounds	48	অন্তর[heart], আংশিক[partial], খোরাক[food], তারিখ[date], পশ্চিম[west], শ্রীমতী[ms], সাহিত্য[literature]
25	-None-	(Couldn't Classified)	774	অকস্মাৎ[bump], অপছন্দ[dislike], কৈফিয়ৎ[apology], তরফ[behalf], তৃপ্তি [satisfaction], বিদ্রূপ[irony]

Table 11: complete POS tag list in Bangla

7.4.3 Translate Bangla word:

As most of the datasets are for English language, it is badly needed to make our own dataset on Bangla language. For this purpose, we have to translate all Bangla words into English words. To translate all Bangla words, we follow two processes, automatic and manual. For automatic process, we use Google translator. We can use many other online translator but we choose it for its accuracy and rich vocabulary. It is being enriched day by day. We translate all words from google translator using python translator library "GOSLAT". This tool is easy to use and very effective. We translate 57000 Bangla words within a moment. In this process, some problem arises.

- We cannot translate all words using google translator
- Some translated words are wrong.
- Some words are translated into phonetic form.

To solve these problem, we check and recheck our whole word list after translation. We solve these errors using manual process. We translate some words manually using bilingual dictionary like Ovidhan and samsad.

7.4.4 Scoring from SentiWordNet :

SentiWordNet is very important for our research. SentiWordNet is used to extend the large and frequently used WordNet resource by sentiment scores. In this manner NLP applications can access both semantic and sentiment information relying on one resource.

Wordnet :

The idea behind WordNet is to create a "dictionary of meaning" integrating the functions of dictionaries and thesauruses. Lexical information is not organized in word forms, but in word meanings which is consistent with the human representations of meaning and their processing in the brain. Besides creating a innovatively organized lexical semantic resource, the researchers aim furthermore to support and promote automatic text analysis for applications in the field of artificial intelligence.

Wordnet Structure and contents :

WordNet contains English nouns, verbs, adjectives and adverbs. They form so called “synsets”, i.e. sets of distinctive cognitive synonyms, which glosses, i.e. descriptions of the synsets with sample expressions or sentences, are attached to. What constitutes the “net”-like structure of WordNet are the links between the synsets. Synsets that have a certain lexical or conceptual relation are linked.

- Nouns can be connected through hyperonymy/hyponymy and meronymy/holonymy relations which can also be inherited. They form a hierarchy which all goes back up to one root. There is also a differentiation between types (common nouns) and instances (persons, entities).
- Verbs are organized via troponym, hypernym and entailment relations
- Adjectives are linked to their antonyms, and relational adjectives point to their related nouns.
- Adverbs make up the smallest group of synsets. They are mostly derived from adjectives and are linked to them via a pertainym relation.

Additionally there are very few cross-POS relations. Morphosemantic links connect words that share the same stem, as for many adverbs and adjectives. Some noun-verb pairs are furthermore annotated for semantic roles. In the current version there are 82115 distinct noun synsets, 13767 for verbs, 18156 for adjectives and 3621 for adverbs, which sums up to 117659 synsets composed by 155287 unique words all in all. The 20-volume Oxford English Dictionary records 171476 words in current use. Estimating the number of words in the English language by this number, WordNet already covers the major part.

SentiWordNet :

The purpose of SentiWordNet the aim of SentiWordNet is to provide an extension for WordNet, such that all synsets can be associated with a value concerning the negative, positive or objective connotation. SentiWordNet 3.0 is the improved version of SentiWordNet 1.0 and publicly freely available for research purpose with a webinterface.

This extension labels each synset with a value for each category between 0.0 and 1.0. The sum of the three values is always 1.0, so each synset can have a nonzero value for each sentiment. but there is no clue in the SentiWordNet regarding which value to pick in what context? The general trend is to pick the highest one but that may vary with context. The following example may illustrate the problem better: the word “*High*” (Positivity: 0.25, Negativity: 0.125 for “*High*” in the SentiWordNet) is attached with a positive (positivity value is higher than the negativity value) polarity in a text but the polarity of that word may vary in any particular use. The word “high” has a positive polarity in the first sentence while the same word has a negative polarity in the second sentence.

Sensex reaches high+

Price goes high-

Actually further NLP techniques are required to disambiguate these types of words. The statistics from the SentiWordNet (English) is presented in Table 4 to understand the big picture that shows how many words are ambiguous and need a special care. There are 6619 lexicon entries in the SentiWordNet where both the positivity and the negativity values are greater than zero whereas the total number of entries in the SentiWordNet (English) is 115424. Therefore, these entries are ambiguous because there is no clue in the SentiWordNet which value to pick in what context? Similarly there are a total of 17927 lexical entries in the SentiWordNet, whose positivity and negativity value difference is less than 0.2. These are also the ambiguous words.

Type	Number
Total Token	115424
Positivity>0 && Negativity>0	6619
Positivity>0 || Negativity>0	28430
Positivity>0 && Negativity=0	10484
Positivity=0 & Negativity>0	*11327*
Positivity − Negativity >=0.2	17927

Table 12: A Closer Look on the Ambiguous Entries of SentiWordNet

Structure of SentiWordNet :

SentiWordNet is a lexical resource for opinion mining. SentiWordNet assigns to each synset of WordNet three sentiment scores: positivity, negativity, objectivity. All words are grouped into set of synonyms called *synset*. SentiWordNet 3.0 and sentiWordNet 1.0 is now available. In a SentiWordNet, every English word has a POS tag, an id number, positivity-negativity score, synonyms and corresponding example. Words are classified into noun,verb,adverb,adjective only.

R 00011093 0.375 0 well#1 good#1 "he slept well"; "the baby can walk pretty good"

Table 13: sentiwordnet structure for a word "*good*"

Here from the table 5, we can see that , R is the tag of word “good”. R stands for adverb, 00011093 is the id number , 0.375 is positivity score , 0 is negativity score , “well” is the synonym of the word “good” and “he slept well”,”the baby can walk pretty good” is tha use or example of that word “good”.

synsets	positive	negative	objective
good#1	0.75	0	0.25
divine#1	0.875	0	0.125
solid#1	0.875	0	0.125
superb#2	0.875	0	0.125
abject#2	0	1	0
pitiful#2	0	1	0
bad#1	0	0.625	0.325
unfortunate#1	0	0.125	0.875

figure 3 : scores for the most positive/negative synsets in SentiWordNet 3.0

SENTIWORDNET visualization:

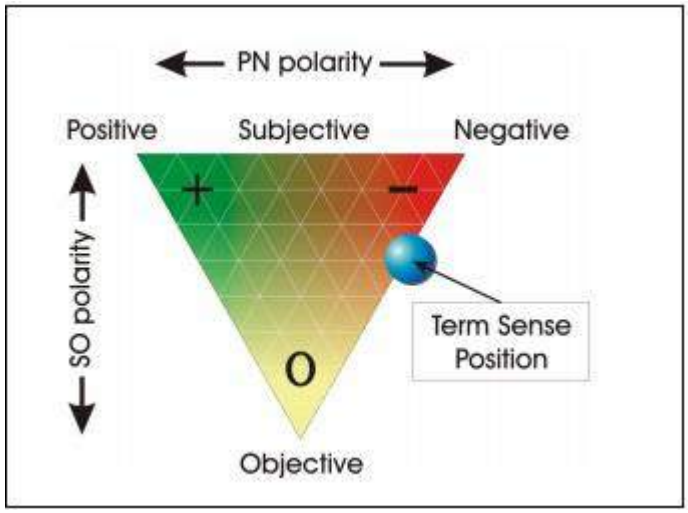
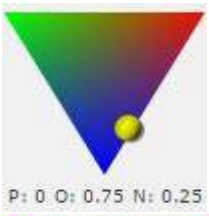
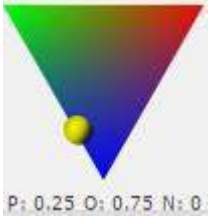


Figure 4: The graphical representation adopted by SENTIWORDNET for representing the opinion-related properties of a term sense.

SENTIWORDNET visualization of the opinion related properties of the term “estimate” :

Noun:

	idea#4 estimation#3 estimate#1 approximation#1 an approximate calculation of quantity or degree or worth; "an estimate of what it would cost"; "a rough idea how long it would take"
	estimation#4 estimate#2 a judgment of the qualities of something or somebody; "many factors are involved in any estimate of human life"; "in my estimation the boy is innocent"
	estimation#1 estimate#3 appraisal#2 a document appraising the value of something (as for insurance or taxation)
	estimate#4 a statement indicating the likely cost of some job; "he got an estimate from the car repair shop"
	estimation#2 estimate#5 the respect with which a person is held; "they had a high estimation of his ability"

SENTIWORDNET visualization of the opinion related properties of the term "SHORT":

Noun:

	short#1 the location on a baseball field where the shortstop is stationed
	short_circuit#1 short#2 accidental contact between two points in an electric circuit that have a potential difference
	shortstop#2 short#3 the fielding position of the player on a baseball team who is stationed between second and third base

Extracting Score:

To make our desirable dataset , we have to find out Bangla sentimental score. To achieve this step, we extract score from SentiWordNet for Bangla words. At first we match Bangla words with corresponding English words based on Parts Of Speech tag. Then we extract positive , negative score of matching words from Sentiwordnet.

Now at this stage, we find all Bangla words, corresponding Bangla POS tag, English word with English POS tag and positive negative, objectivity score of words.

word	tag	eng	sentiTag	positive	negative	neutral
Filter	Filter	Filter	Filter	Filter	Filter	Filter
ছাঁট	JJ	clipping	n	0	0	1
মুদ্রাপ্রস্তুতকারী	NN	coiner	n	0	0	1
মুদ্রাপ্রস্তুতকারী	NN	coiner	n	0	0	1
মুদ্রাপ্রস্তুতকারী	NN	coiner	n	0	0	1
স্বচ্ছতা	NN	transparency	n	0	0	1
স্বচ্ছতা	NN	transparency	n	0.5	0.125	0.375
স্বচ্ছতা	NN	transparency	n	0.125	0	0.875
অমলতা	NN	clarity	n	0.375	0.125	0.5
অমলতা	NN	clarity	n	0.5	0.125	0.375
ঝিঁঝি	NN	cricket	n	0	0	1
ঝিঁঝি	NN	cricket	n	0	0	1
ঝিঁঝি	NN	cricket	v	0	0	1
পোকার	NN	insect	n	0	0	1
পোকার	NN	insect	n	0	0.25	0.75

Figure 5: final dataset

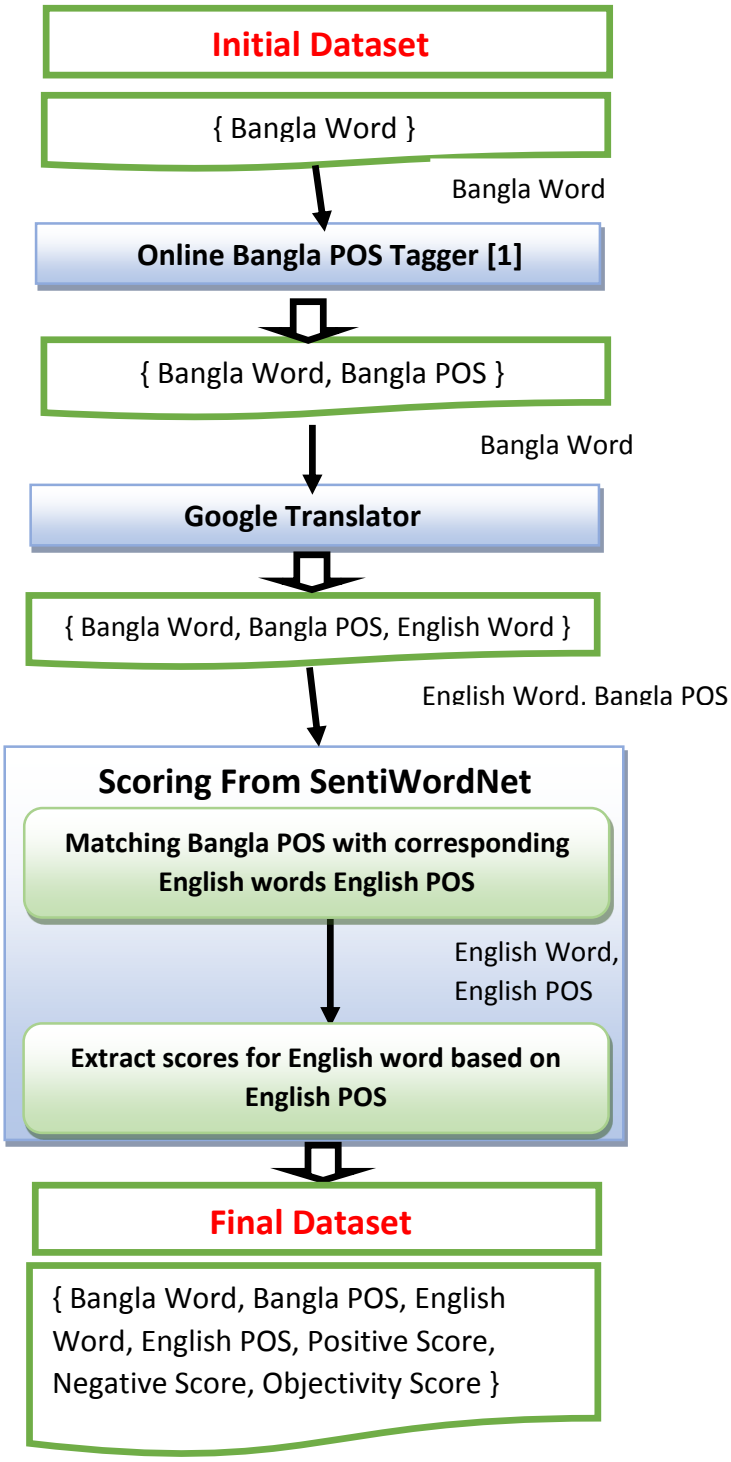


Figure 6: step by step process of making dataset

8: Working Process:

8.1 Preprocessing:

The input text needs to be preprocessed to get the most usable tokens for the calculation of sentiment analysis. By preprocessing, it removes noisy symbols, tags, links which are not expressing any sentiment. Our preprocessing includes four steps – separating emoticons, tokenization, normalization and token classification.

From an input text, every words go through a tokenizer after checking whether it is an emoticon or not. We use python NLTK Tokenizer[] for tokenization. NLTK tokenizer splits the emoticons which are actually a collection of punctuation marks into single characters. That is why we keep a check for every words before tokenization to identify emoticons. We use regular expression to identify an emoticon.

After tokenization, we perform normalization process for every token. Normalization includes the removal of URL, punctuation marks for both English and Bangla and also special characters like white space, new lines etc.

After normalization process, we identify whether the token is in the form of Bangla or English. Based on the classified token [Bangla/English token], the token will be searched in the database if it exists or not to extract its sentimental values.

An example of preprocessing:

Input Text: আজকের বৃষ্টি ভেজা আবহাওয়ার জন্য Match টি হল না...! :(

Extract Emoticon		:(
Tokenization		“আজকের” ,“বৃষ্টি , “ভেজা” ,“আবহাওয়ার” ,“জন্য” ,“Match” ,“টি”“হল” ,“না” ,“.”, “.”, “.”, “!”
Normalization		“আজকের” ,“বৃষ্টি , “ভেজা” ,“আবহাওয়ার” ,“জন্য” ,“Match” ,“টি”“হল” ,“না”
Token Classification	Bangla	“আজকের” ,“বৃষ্টি , “ভেজা” ,“আবহাওয়ার” ,“জন্য” ,“টি”“হল” ,“না”
	English	“Match”
	Emoticon	:(

Table 14: Preprocessing steps for a Bangla sentence:

8.2 Searching in Database:

After the preprocessing step, tokens are searched in database to get their positive, negative values. Token searching in the database can be expressed in the following steps:

Firstly, a token is checked if it is already searched in DB or not. If it is already searched in DB then the scores of that token is taken from a list called “Searched Token”. The list contains the tokens scores sequentially which are already found by previous searching. If the token is not already searched then a query is executed in DB based on the classification of that token.

Secondly, a token can exist in the database or not. If the token is found in database then its positive negatives values are extracted and saved into the “Searched Token” list. If the token is not found in database then an approximate word searching is performed.

Thirdly, by the approximation searching if the token is found in database, its positive negatives scores are extracted and saved into the “Searched Token” list. If the token is not found by the approximation search, the word is listed as “Unused Words”. Unused words have no effect in the calculation of sentimental analysis.

8.3 Approximate Word Searching

In approximate word searching, a special file searching is performed for matching an approximate word to the token. There are different files containing all the words started with same alphabetic letter. The token is searched in the file which name is corresponding English token’s first letter. For Bangla token, it is searched in the file which name is converted into English phonetic form of the Bangla token’s first letter. When there is wrong or miss composing of a word, basically most of the time the initial alphabet of that word never miss typed. Based on this we designed our file searching system for searching an approximate word. We use Python “difflib” library for the approximation searching process with at least 94% matching ratio. If the token is found after file searching, its positive and negative scores are taken from database and saved into “Searched Token” list.

For example one of the files name is “ka.txt” which contains all the words started with Bangla alphabet “ক” .

Let us consider a miss composed word is “প্রশমণ” where correct word is “প্রশমন” . When the word “প্রশমণ” will not be found in Database, then it will start searching for an approximate word with a matching ratio of 94% and above. For the word “প্রশমণ” it will start looking at “pa.txt” file where all the words started with ”প” . This separate file searching system makes the approximate word searching too fast and more accurate. After the searching it finds the word “প্রশমন” with maximum matching ratio (94.4%)

If the token is not found after file searching, and it is an English token then it is saved into "Unused Words" list. If the token is a Bangla token then we are checking for if the token has any negative suffix at the end of the word like "না", "নি". If the token contains any negative suffix then the token is splitted into two parts. First part contains the word without "না", "নি" which is called prefix word and the last part "না"/"নি" is called negative suffix word. The prefix word is searched in database for extracting its sentimental values. If the prefix word is found in database then both scores of prefix and negative suffix words are taken and sequentially saved into "Searched Token" list. If the prefix word is not found in database then both the prefix and suffix words are discarded and saved in the "Unused Words" list.

For an example, let us consider a Bangla word "খারাপনা".

This word doesn't exist in our database. So it cannot be found in file searching. After that it will be checked if it has any negative suffix like "না"/"নি". It has a negative suffix "না" in end of the word. So it will be splited into prefix word "খারাপ" and negative suffix word "না". Now the prefix word will be searched into database. As the word "খারাপ" exists in our database, so the scores for both words"খারাপ" and "না" will be taken and saved sequentially in "Searched Token" list.

Diagram of Approximate Word Searching:

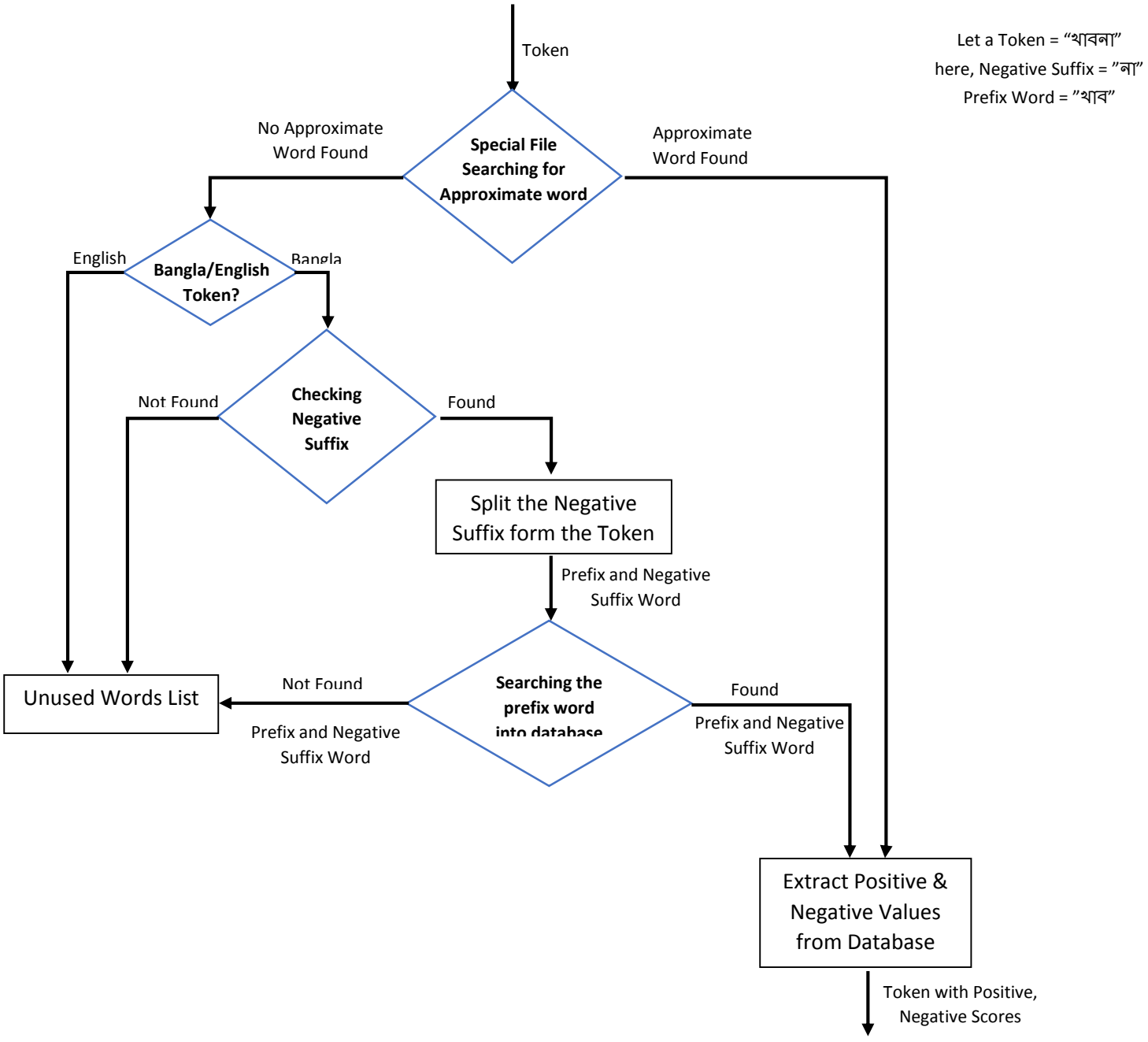


Figure 7: Approximate Word Searching

8.4 Sentimental Analysis:

The tokens scores from the "Searched Token" are taken sequentially for our final calculation of sentimental analysis. We consider tokennegative as the negative value of that token is greater than the positive value of it, and tokenpositive as the positive value of that token is greater than the negative value of it. **next_token** means the consecutive token of the current token's which is using for calculation. **Fixed_Negative_Words** are 'না','নই','নেই','নাই','নয়','নউ','নও','নো','নোউ','নোও','নাহ'.We use the following algorithm:

if **next_token** := **Fixed_Negative_Words**

if token$_{negative}$

total_positive_score = total_positive_score + token_negative_score+next_token_negative_score;

if token$_{positive}$

total_negative_score = total_ negative_score + token_postive_score+next_token_positive_score;

else if **next_token** := **token AND Token_POS_Tag= Adjective**

if token$_{negative}$

total_positive_score = total_positive_score + 2*token_positive_score;
total_negative_score = total_ negative_score + token_negative_score;

if token$_{positive}$

total_positive_score = total_positive_score + token_positive_score;
total_negative_score = total_ negative_score + 2*token_negative_score;

else

total_positive_score = total_positive_score + token_positive_score;
total_negative_score = total_ negative_score + token_negative_score;

Finally comparing the total_positive_score with total_negative_score the output sentiment is determined. If the total positive score is greater than the total negative score then the output is considered as positive sentiment and vice versa. If the two scores are equal then the input text is considered as neutral sentiment.

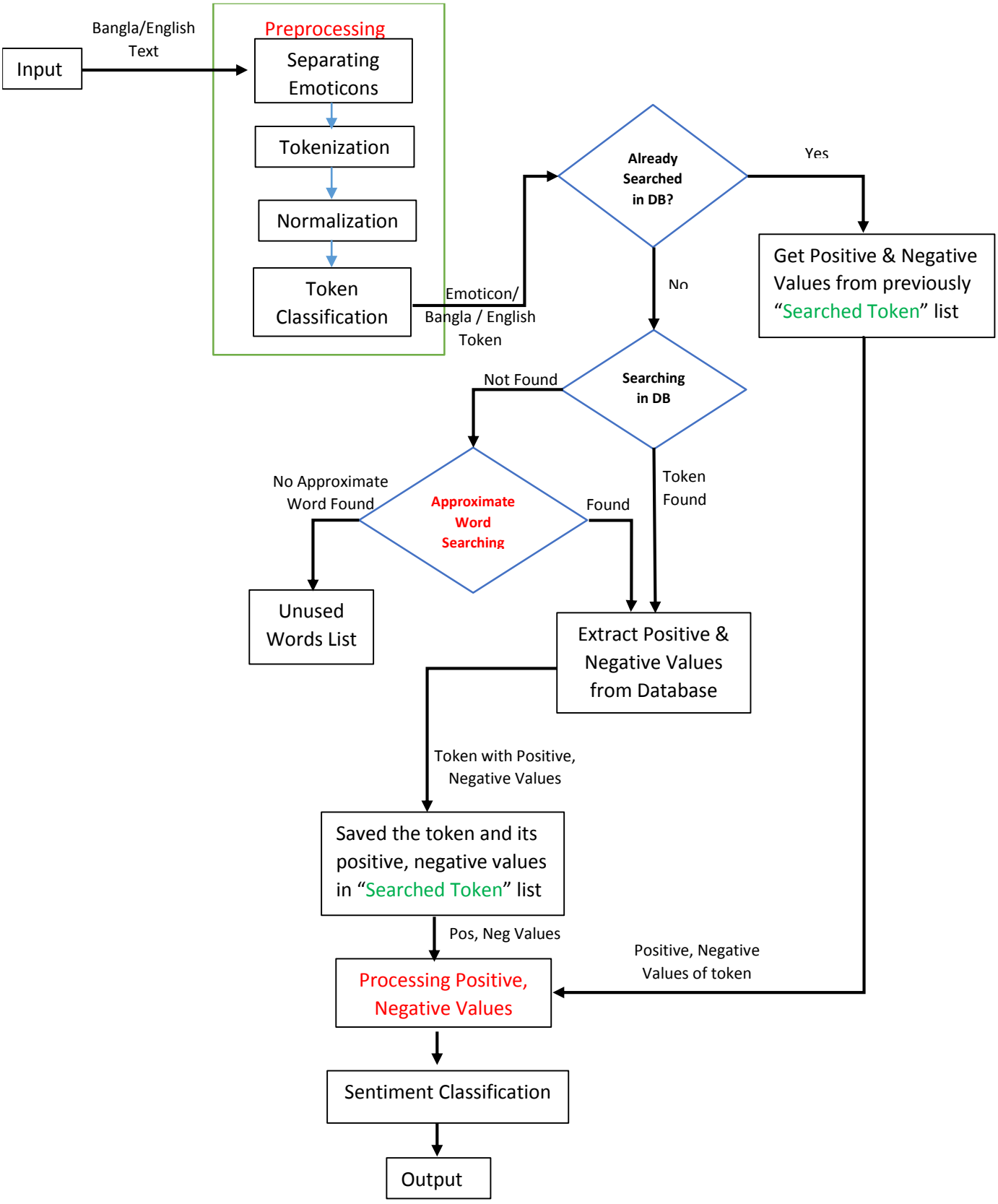
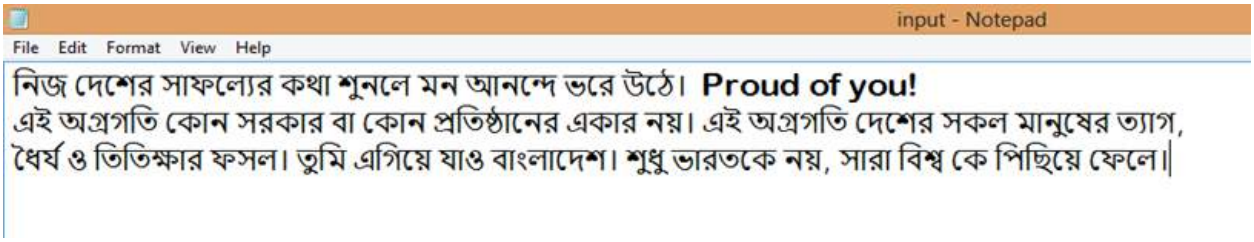


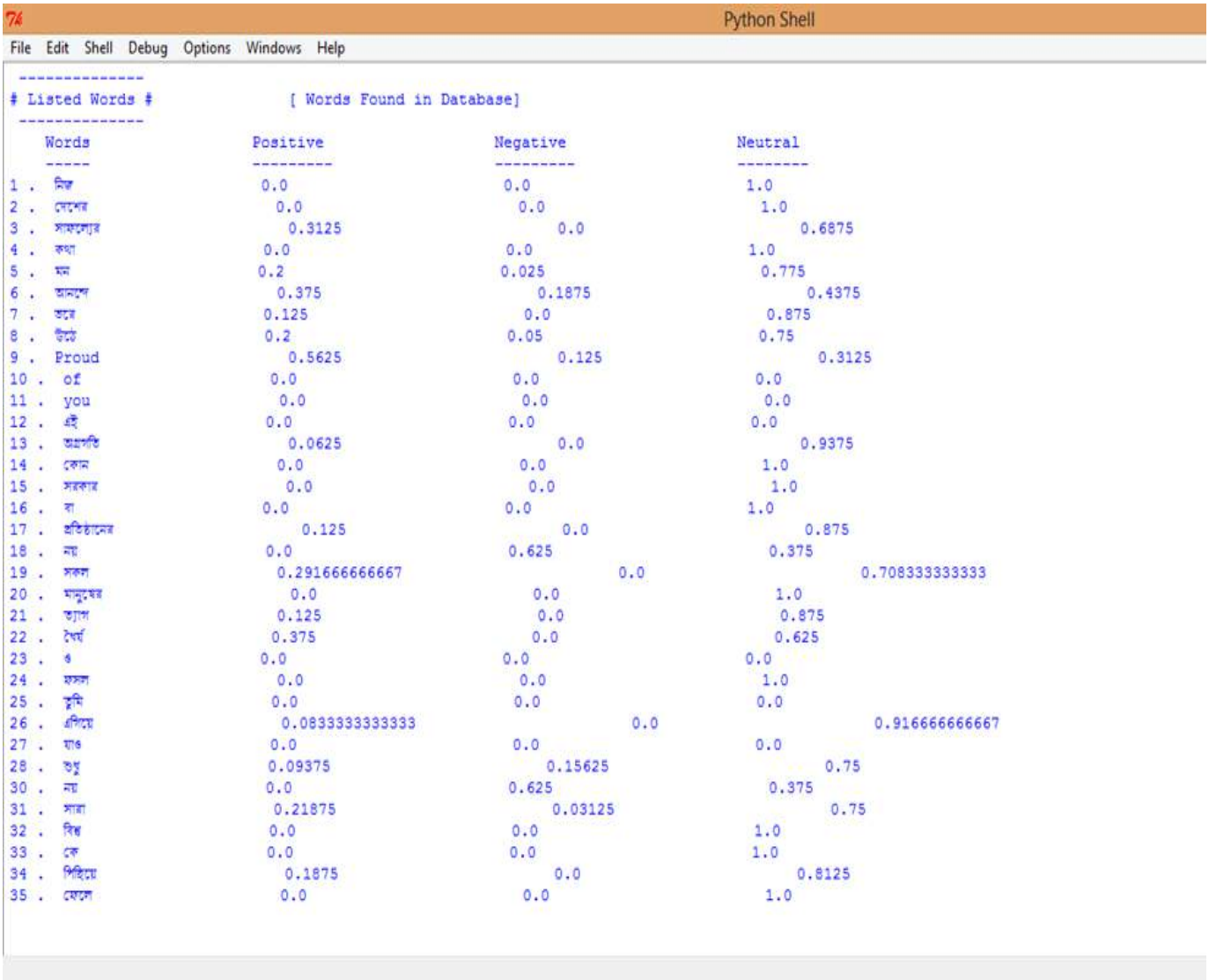
Figure 8: Main Diagram

8.5 An Example of the Whole Process

Input Text:



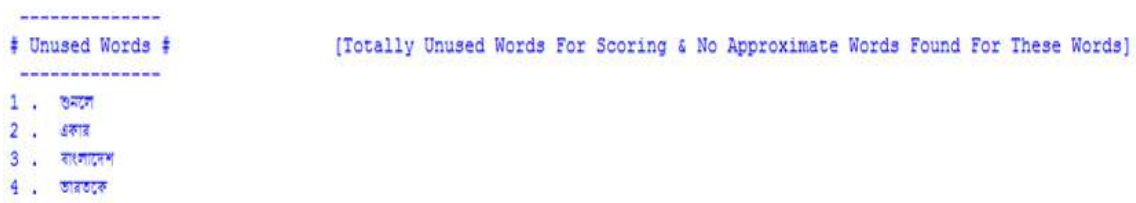
Listed Words [Found words in the Database]:



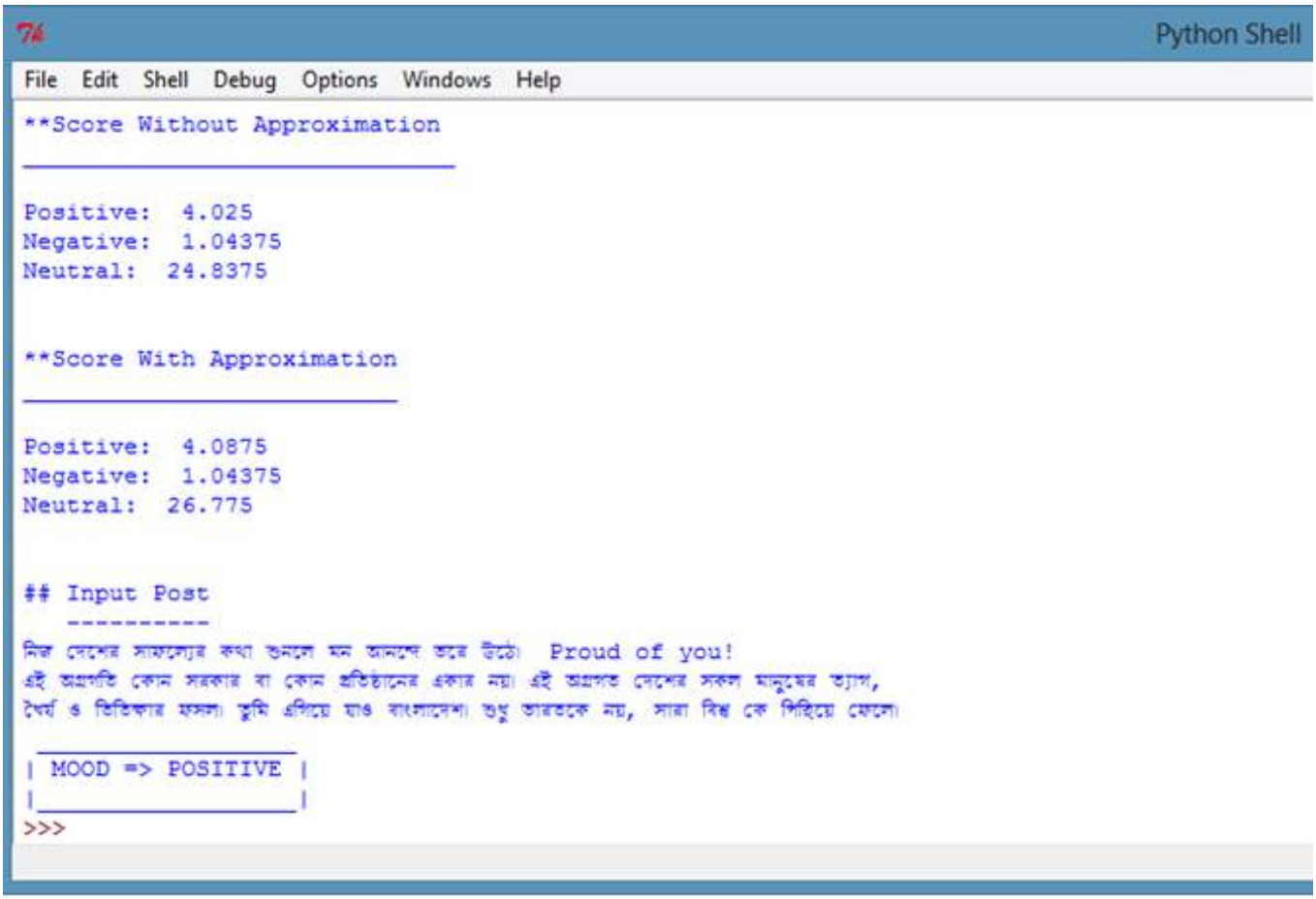
Unlisted Words [Not Found in the Database]:

Approximate Words [Approximating words form Unlisted Words]:

Unused Words [Totally Unused Words for the Final Calculation]:



Final Output:



In this case of input text, the final output is expressing positive sentiment. Here the scores of the words are calculated using our algorithm for sentiment analysis. Both of the results with or without the approximate words are shown in the output. Comparing the positive and negative scores, the sentiment is determined. Here we can see that the positive score in both results is greater than the negative score. So the input text is determined as an example of positive sentiment.

Graphical Representation of the Output Sentiment:

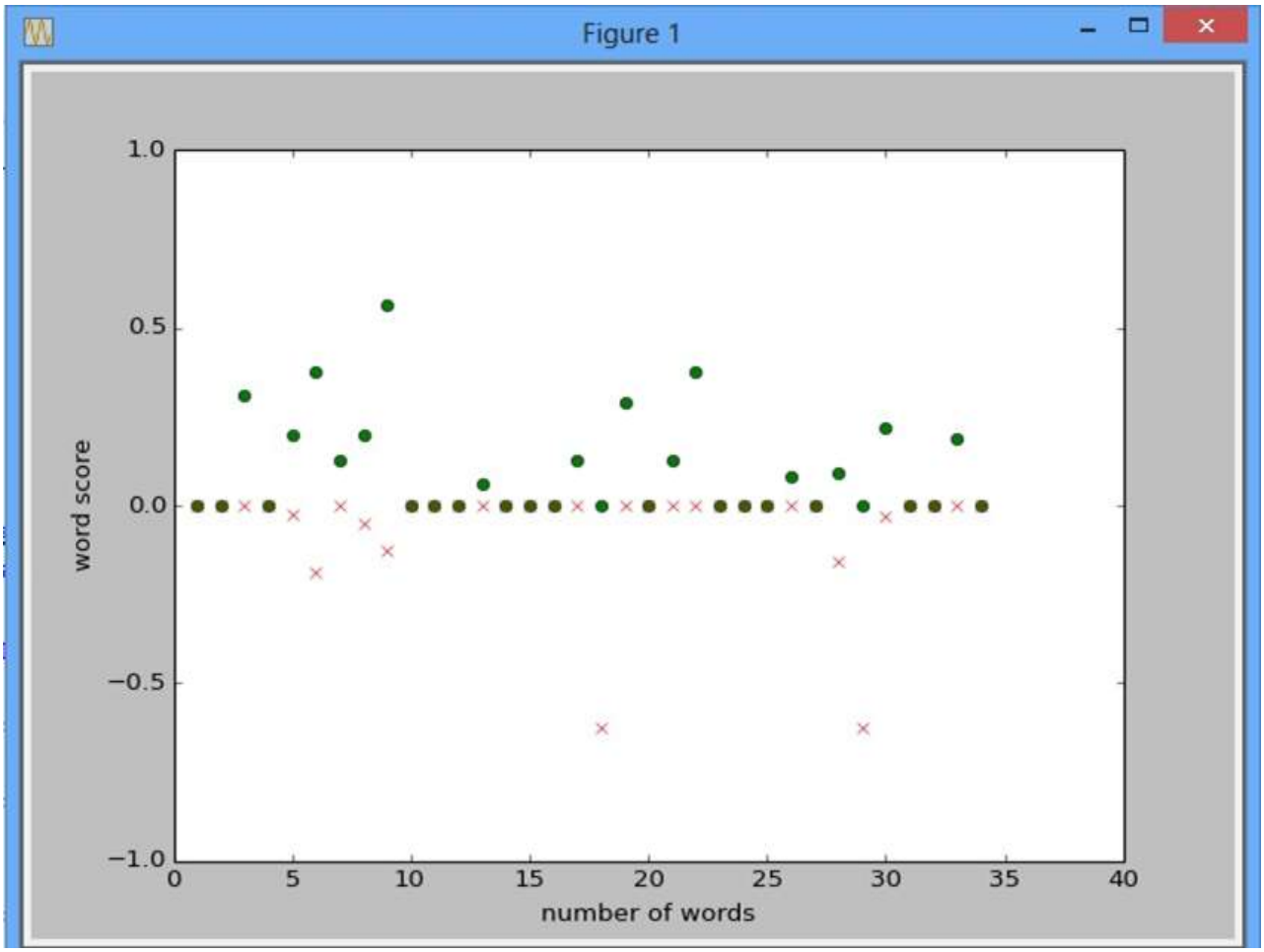


Figure 9: Graphical representation of the output sentiment

- Graph is based on Positive and Negative Scores
- Positive Scores are Green Dots and Negative Scores are Red Cross

Table 15: Example of some Adjectives from our Database

Serial No.	Bangla Word	English Word	Positive Score	Negative Score
1	অস্থির	unstable	0.1875	0.375
2	অস্থিরগত	planetary	0.0	0.208333333333333
3	অস্থিরচিত্ত	frivolous	0.0	0.75
4	অস্থিরীকৃত	indeterminate	0.2	0.35
5	অস্থিসংক্রান্ত	osseous	0.0	0.0
6	অস্পর্শিত	untouched	0.125	0.5
7	অস্পষ্ট	vague	0.3125	0.3125
8	অস্পৃশ্য	untouchable	0.0	0.46875
9	অস্পৃষ্ট	untouched	0.125	0.5
10	অস্ফুট	indistinct	0.25	0.75
11	কৃপণোচিত	ungenerous	0.25	0.1875
12	কৃপাপ্রার্থীসুলভ	ingratiating	0.4375	0.0
13	কৃশতাপ্রাপ্ত	attenuate	0.0	0.375
14	কৃষকসুলভ	rustic	0.0625	0.0
15	কৃষি-সংক্রান্ত	agronomic	0.0	0.0
16	কৃষিজাত	agricultural	0.0	0.0
17	কৃষ্টিগত	cultural	0.0	0.0
18	কৃষ্টিসম্পর্কিত	cultural	0.0	0.0
19	কৃষ্ণাভ	blackish	0.0	0.125
20	কেঁদ্রগত	pivotal	0.75	0.25
21	জগদ্-বহির্ভূত	spiritual	0.125	0.0833333333333333
22	জঘণ্য	pitiful	0.152666666666667	0.847333333333333
23	জঘন্য	shocking	0.1875	0.3125
24	জঙ্গলবাসিসুলভ	jungly	0.0	0.0
25	জনগণসংক্রান্ত	folksy	0.0625	0.1875
26	জননীসংক্রান্ত	maternal	0.0	0.0
27	জননীসুলভ	maternal	0.0	0.0
28	জনপ্রি	exoteric	0.25	0.0
29	জনবক্তাসুলভ	oratorical	0.0	0.125
30	জনমানবশূন্য	uninhabited	0.0	0.5
31	দাতব্য	charitable	0.541666666666667	0.125
32	দানবতুল্য	fiendish	0.125	0.875
33	দাম্পত্য	bridal	0.0	0.0

Table 16: Example of some Nouns from our Database

Serial No.	Bangla Word	English Word	Positive Score	Negative Score
1	অকৃতদার	celibate	0.0	0.0
2	অকৃতিত্ব	failure	0.104166666666667	0.354166666666667
3	অকৃত্রিমতা	legitimacy	0.25	0.1875
4	অকৃপণ	liberal	0.0	0.0
5	অকৌশল	quarrel	0.0625	0.125
6	অক্কা	god	0.0625	0.0
7	অক্কাপ্রাপ্তি	death	0.05	0.275
8	অক্টোপাস	octopus	0.0	0.0
9	অক্রম	confusion	0.09375	0.28125
10	অক্রিয়তা	torpor	0.0625	0.1875
11	কোষাগার	treasury	0.0625	0.0
12	কোষাধ্যক্ষ	treasurer	0.0	0.0
13	কোষ্ঠবদ্ধতা	constipation	0.0	0.375
14	কোষ্ঠশুদ্ধি	motion	0.0	0.0625
15	কোষ্ঠী	horoscope	0.0	0.0
16	কোহল	spirit	0.21875	0.03125
17	কৌঁসুলি	counsel	0.0	0.0
18	কৌচ	daybed	0.0	0.0
19	কৌটা	casket	0.0	0.0
20	কৌতুক	trick	0.15	0.1
21	পাকস্থলীর	stomach	0.0416666666666667	0.0833333333333333
22	পাকাইয়া	volume	0.0	0.1875
23	পাকাইয়া-রাখা	volume	0.0	0.1875
24	পাকানো	roll	0.0416666666666667	0.125
25	পাকাশয়গহ্বর	pit	0.0	0.25
26	পাকড়	seizure	0.125	0.0
27	পাখনা	fin	0.0	0.0
28	পাখা	fan	0.0625	0.0
29	পাখার	wings	0.0	0.0
30	পাখি	bird	0.0416666666666667	0.125
31	পাখিঘর	aviary	0.0	0.0
32	পাখিবিশেষ	snipe	0.0	0.0
33	পাখী	catcher	0.0	0.0

Table 17: Example of some Adverbs from our Database

Serial No.	Bangla Word	English Word	Positive Score	Negative Score
1	প্রায়	about	0.25	0.125
2	প্রায়ই	often	0.208333333333333	0.0
3	প্রয়োজনবশতঃ	perforce	0.0	0.0
4	ফলতঃ	consequently	0.0	0.0
5	বাহ্যতঃ	apparently	0.5	0.0625
6	বিশেষত	especially	0.0	0.0
7	বিশেষতঃ	especially	0.0	0.0
8	বিশেষভাবে	specifically	0.375	0.0
9	ব্যাপকভাবে	widely	0.0	0.0
10	ভালভাবে	well	0.395875	0.072875
11	ভালোভাবে	better	0.4375	0.0
12	মাঝে	occasionally	0.0	0.0
13	মাত্র	only	0.09375	0.15625
14	মূলতঃ	mainly	0.0	0.0
15	মোটামুটি	rough	0.0	0.25
16	শুধু	only	0.09375	0.15625
17	সত্যিই	really	0.4375	0.0625
18	সম্ভবত	likely	0.0	0.0
19	সম্ভবতঃ	presumably	0.0	0.0
20	সযত্নে	diligently	0.375	0.0
21	সর্বত্র	everywhere	0.375	0.0
22	সর্বাগ্রে	foremost	0.0	0.0
23	সহজেই	easily	0.291666666666667	0.0
24	সাধারণত	generally	0.0	0.1875
25	সাধারণতঃ	usually	0.0	0.0
26	সাধারণভাবে	generally	0.0	0.1875
27	সারতঃ	essentially	0.5	0.0
28	সুবিধামতো	conveniently	0.125	0.0
29	স্থায়ীভাবে	permanently	0.25	0.0
30	স্বচ্ছন্দে	freely	0.375	0.0
31	স্বভাবতঃ	naturally	0.0625	0.125
32	স্বভাবতই	naturally	0.0625	0.125
33	স্বাধীনভাবে	freely	0.375	0.0

Table 18: Example of some Verbs from our Database

Serial No.	Bangla Word	English Word	Positive Score	Negative Score
1	অপ্রকৃতভাবে	misunderstand	0.0	0.625
2	অপ্রযত্ন	neglect	0.0416666666666667	0.125
3	অপ্রস্তুতে	countenance	0.0	0.0
4	অপ্রীতিভাজন	stink	0.125	0.8125
5	অবগাহন	bath	0.0	0.0
6	অবগুণ্ঠন	veil	0.0	0.125
7	অবজ্ঞাচ্ছলে	belittle	0.125	0.166666666666667
8	অবজ্ঞায়	scorn	0.0	0.3125
9	অবধে	range	0.15	0.125
10	অবমূল্যায়ন	underestimate	0.0416666666666667	0.125
11	প্রণোদন	motion	0.0	0.0
12	প্রণয়	romance	0.21875	0.03125
13	প্রণয়ভাবে	mouse	0.0625	0.0
14	প্রতিদানে	reciprocate	0.0	0.0
15	প্রতিবাদে	protest	0.0	0.0
16	প্রতিবিধান	redress	0.0	0.125
17	প্রতিবেদন	report	0.125	0.03125
18	প্রতিষ্ঠানভবন	institute	0.0	0.0
19	প্রতিস্থান	replace	0.0	0.25
20	প্রতীক্ষায়	wait	0.0	0.125
21	প্রত্যক্ষজ্ঞানসম্পন্ন	witness	0.0625	0.0
22	প্রত্যবর্তন	retrograde	0.0	0.25
23	যুক্তিবলে	contest	0.0	0.0
24	যুক্তিসহভাবে	fair	0.0	0.0
25	যুদ্ধার্থে	war	0.0	0.0
26	যুদ্ধে	war	0.0	0.0
27	যোগদান	join	0.0	0.0
28	যোগান	provide	0.0625	0.0
29	যোগ্যতাদান	habilitate	0.0	0.0
30	যোগ্যতাহীন	unfit	0.0	0.375
31	যৌনসঙ্গমে	sex	0.25	0.0625
32	যৌনসম্পর্কে	sex	0.25	0.0625
33	যোগদান	join	0.0	0.0

9: Future Works

9.1 Extract Scores from Banglish Sentence: Our next target is to extract scores from Banglish words. Banglish words mean Bangla word but written in English phonetic form. In example," ভালো লাগা কিছু মুহূর্ত" which can be written "*Valo laga kichu muhurto*". We have to extract sentimental scores from these form. For this purpose, we use Avro Phonetic Library in Python. This library converts Banglish sentences into Bangla sentences.

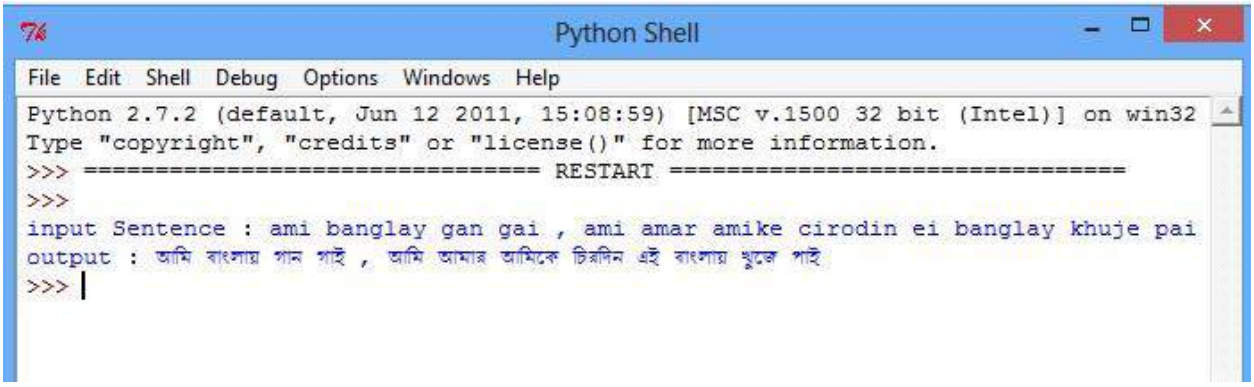


Figure 10: Banglish to Bangla sentence conversion Using Python

9.2 Dataset accurate: Our Dataset contains only nearly 57000 Bangla words. But Bangla is a morphologically rich language. We can accurate our sentiment extraction process by adding stemming Bangla words or adding more words into the dataset. More words in the database, more the accuracy. Adding more and more words is the easiest way to find more accurate result than adding stemming rules.

9.3 Synonyms: A word may have various types meaning and if meaning differs, then scoring is not same. "ভালো দিন দেখে কাজটা সেরে ফেলো" and "অতি ভালো ভালো না", here the word "ভালো" bears different meaning. It is difficult to extract exact score without sense. If we want to extract exact scores, then we have to add synonyms of the word.

9.4 Own scoring methods: We find out scores of our Bangla words from Sentiwordnet. But as Bangla is a rich language, we cannot find scores of some words from sentiwordnet. So we are working to find our own scoring methods.

9.5 More rules for score calculation: To improve the result of our sentiment analysis process, we have many opportunities for development. We can work with repeated words like "হাসি হাসি মুখ", "কাঁদো কাঁদো চেহারা" etc. Again we have to consider 'উপসর্গ","অনুসর্গ" and can add some rules for these. In example, "অনাচার", "অসামাজিক","কুকর্ম","নিষ্পাপ" these are some special words by adding "উপসর্গ"

“অনা”,”অ”,”কু”,”নি”. Here , “সামাজিক” is a positive word , but when we add “অ” with this, “অসামাজিক” is a negative word. In the same way, “পাপ” is a negative word, but when we add “নি”, “নিষ্পাপ” is a positive word.

There are some Bangla and English words in our database without sentimental scores. As we scored the words using SentiWordNet some English words score couldn’t found. We will work on scoring those words without scores in our database.

Serial No.	Bangla Word	English Word	Bangla POS Tag
1	অ-ঈর্ষিত	unenvied	JJ
2	অ-কার্যোপযোগী	non-serviceable	NN
3	অ-কুশল	non-well-being	NN
4	অ-ক্ষতিগ্রস্ত	non-damaged	JJ
5	অ-চলচ্চিত্রায়িত	unscreened	JJ
6	উদ্ভাসন	brightening	VM
10	উদ্ভিদে	plants	VM
11	উদ্ভিদের	plants	NN
14	উদ্ভিন্নকৈশোর	juvenescent	NN
16	এদিক-ওদিকে	back-side	VM
17	এদিকে-ওদিকে	meanwhile-toss	VM
18	এদিক্	hitherwards	CC
19	এনে	the	VM
20	এপার	citizens near	NN
21	এপাশ	this side	NN
22	এপ্রিকট	apricots	JJ
24	এবং	and	CC
25	এবংবিধ	like this	NN
27	কাপ্তেনি	foppery	NN
28	কাপড়-কাটা	clothes-cut	NN
33	কাব্য-পংক্তি	belles-string	NN
34	কাব্য-সঙ্গীত	poetry-music	JJ

10: References:

1. https://www.alsintl.com/blog/most-common-languages/
2. http://dailynewsdig.com/top-ten-spoken-languages-world-2014/
3. http://www.prothom-alo.com/
4. http://bdnews13.blogspot.com/2011/12/top-10-bangladeshi-websites.html
5. 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.
6. N. Kobayashi, K. Inui, K. Tateishi, and T. Fukushima. Collecting evaluative expressions for opinion extraction. In Proceedings of IJCNLP 2004, pages 596–605, 2004.
7. 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.
8. 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.
9. 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.
10. 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.
11. 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.
12. 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
13. 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.
14. 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.

15. 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.
16. https://en.wikipedia.org/wiki/Sentiment_analysis
17. 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.
18. Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. Technical report, Stanford.
19. 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.
20. 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.
21. 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.
22. http://csea.phhp.ufl.edu/media/anewmessage.html
23. http://www.wjh.harvard.edu/~inquirer/
24. http://mpqa.cs.pitt.edu/opinionfinder/
25. http://sentiwordnet.isti.cnr.it/
26. http://wndomains.fbk.eu/wnaffect.html
27. 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.
28. 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.

29. FA Nielsen. 2011. A new ANEW: Evaluation of a word list for sentiment analysis in microblogs.
30. A Kennedy, D Inkpen. 2006. Sentiment classification of movie reviews using contextual valence shifters. In Computational Intelligence, Wiley Online Library
31. Eugenio Martínezcámara, M. Teresa Martínvaldivia,L. Alfonso Ureñalópez and A Rturo Montejoráez. Sentiment analysis in Twitter. Natural Language Engineering.
32. 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.
33. 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.
34. 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.
35. http://www.ling.gu.se/~lager/mogul/brill-tagger/index.html
36. http://www.openmind.org/
37. http://trec.nist.gov/pubs/trec17/papers/BLOG.OVERVIEW08.pdf
38. http://saaip.org/
39. http://cs.stir.ac.uk/~eca/sentics
40. http://cs.stir.ac.uk/~eca/commansense
41. http://en.wikipedia.org/wiki/Paul_Ekman#Emotion_classification
42. http://conceptnet5.media.mit.edu/
43. http://www.amitavadas.com/sentiwordnet.php
44. http://www.lsi.upc.edu/~nlp/SVMTool/
45. http://ltrc.iiit.ac.in/showfile.php?filename=downloads/shallow_parser.php
46. https://dev.twitter.com/docs/api/1.1
47. http://nltk.org/api/nltk.tokenize.html
48. http://nltk.org/api/nltk.tag.html
49. https://github.com/ankur-india/bangla_pos_tagger
50. http://ltrc.iiit.ac.in/analyzer/bengali
51. 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.

52. http://sentiment.christopherpotts.net/code-data/happyfuntokenizing.py
53. http://leebecker.com/resources/semeval-2013/
54. http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html#lexicon
55. 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.