Sentimental Analysis (Bengali)

Adrija Roy
Abhishek Anand Singh

Abstract

Sentiment analysis has received great attention recently due to the huge amount of user-generated information on the microblogging sites, such as Twitter [1], which are utilized for many applications like product review mining and making future predictions of events such as predicting election results. Much of the research work on sentiment analysis has been applied to the English language, but construction of resources and tools for sentiment analysis in languages other than English is a growing need since the microblog posts are not just posted in English, but in other languages as well. Work on Bangla (or Bengali language) is necessary as it is one of the most spoken languages, ranked seventh in the world. In this paper, we aim to automatically extract the sentiments or opinions conveyed by users and then identify the overall polarity of texts as either negative or positive. We used HMM to perform POS tagging and SVM classifier for sentiment analysis.

Introduction

Bangla (or Bengali) language is one of the most spoken languages in the world. Its use is becoming more prevalent with the recent growth of online microblogging sites, where users can easily share their views and opinions on varying issues of interest, such as politics, religion, economics, business, and entertainment, using very short text. This results in huge volumes of user-generated information on the microblogging sites, which are utilized for many applications._In recent years, microblogging sites have become a very popular source for publishing a huge amount of user-generated information. One of the unique characteristics of these microblogging sites is that the messages that are posted by the users are short in length and users publish their views and opinions on different topics such as politics, religion, economics, business, and entertainment. These huge volumes of user-generated information on the microblogging sites are utilized for many applications. Product review mining is one such application where potential consumers go through the opinions expressed by previous consumers on different sites before acquiring a particular product or service, while companies analyze the feedbacks on different products or services posted by consumers on these sites to gain knowledge about which products or services to sell more and which should be improved. These microblogging sites are also used as a source of data for making future predictions of events, such as predicting election results. Here, we are not talking about going through just one or two user messages on a particular product or service and making a decision on that. Instead, millions of messages that are posted daily on the microblogging sites need to be checked, all the relevant posts for that product or service need to be extracted, different types of user opinions need to be analyzed, and finally the user opinions and feedbacks need to be summarized into useful information. For a human being, this is a very tedious and time-consuming work. This is where sentiment analysis comes in use. Sentiment analysis or opinion mining is the automatic extraction of opinions, emotions, and sentiments from texts. Sentiments, opinions, and emotions are subjective impressions and not facts, which are objective or neutral. Through sentiment analysis, a given text can be classified into one of the three categories - positive, negative, or neutral. Sentiment analysis of texts can be performed at different levels like - document, sentence, phrase, word, or entity level.

Much of the research work on polarity classification of Microblog posts has been implemented on the English language, but construction of resources and tools for sentiment analysis in languages other than English is a growing need since the microblog posts are not just posted in English, but in other natural languages as well. Work on other languages is growing, including Japanese , Chinese , German , and Romanian .

In this paper, we aim to automatically extract the sentiment or polarity conveyed by users from Bengali sentences.

Table 1 shows examples of some Bangla tweets with expressed user sentiments.

Sample Bangla Tweets @Kd হুম Samsung Netbook এর performance তুলনামূলকভাবে ভাবলাই মবন হবে। (@Kd yes performance of Samsung Netbook seems comparatively good.)
৬০,০০০ টাকার মমা◌◌াইল ম ◌াবন Bluetooth নাই!!! তাইবল মেই আবেল হাবত না মরবে ে◌◌াওনাটা ভালা। #iPhone #Apple (60,000 taka mobile phone doesn’t have Bluetooth!!! Then instead of keeping that apple in hand it’s good to eat it. #iPhone #Apple)
আমার মনটুেক মেবক মাবে মাবে মৃদুগুড়গুড় আওয়াজ েআবতবে। ঘটনা কক? :S (Sometimes a mild rumbling sound comes from my netbook. What’s the matter? :S)
@nl ে◌যামোঙ'এর ডুওে...কমউকজক কেবেমটা ◌াটা ◌াকট!...#superb#gadget (@nl samsung’s duos...music system is awesome!...#superb#gadget)

Related Work in English

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

Machine learning methods use classifiers that learn from the training data to automatically annotate new unlabeled texts with their corresponding sentiment or polarity. [3] 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 [4]. 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%. [5] follow the same procedures as [4] 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, 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 [6] use 50 hashtags and 15 emoticons as sentiment labels to train a supervised sentiment classifier using the K Nearest Neighbors (KNN) algorithm. In [18], the authors implement a 2-step sentiment detection framework by first distinguishing subjective tweets from non-subjective tweets and then further classify the subjective tweets into positive and negative polarities. The authors find that using meta-features (POS tags) and tweet-syntax features (emoticons, punctuations, links, retweets, hashtags, and uppercases) to train the SVM classifiers enhances the sentiment classification accuracy by 2.2% compared to SVMs trained from unigrams only. Although supervised machine learning methods have been widely employed and proven effective in sentiment classification, they normally depend on a large amount of labeled data, which is both time consuming and labor intensive work.

POS TAGGING

POS tagging for Bengali language has been a task approached in multiple ways in the past by researchers all over the community given the popularity of the language in the Asian countries. We tried to approach the problem ourselves, using NLTK and HMM to learn a model that can accurately classify Bengali words.

In POS tagging our goal is to build a model whose input is a *sentence*, for example –

সত্যেন্দ্রনাথ প্রথম বিলেত থেকে ফিরলেন
।
NP ALC NC PP VM
PU

We use a supervised learning approach and thus have dataset of approximately 4600 training examples along with their corresponding POS tags. We train an NLTK's Hidden Markov model over the training dataset. The model, even with a small dataset, achieved an accuracy of 64% when applied to a similar test data set coming from twitter. Due to a lack of big enough dataset, the bigram and trigram HMM's gave sparse feature vectors and thus a lower tagging accuracy.

LEXICON

Bangla as opposed to English doesn't have a rich resource repository online and thus, when we decided to perform sentiment analysis with the help of polarity lexicons, we had some troubles finding a freely available lexicon list for Bengali language. One approach to creating a bangla lexicon was to use existing English lexicon in Bengali language using bilingual dictionary. Apart from this, SentiWordNet was a good NLP tool which can easily provide us with a sentiment polarity of a word. We decided to go ahead with a hybrid approach, starting with simple translation of English lexicons to their Bengali counterparts and using SentiWordNet to further assess the accuracy of lexicon polarity results.

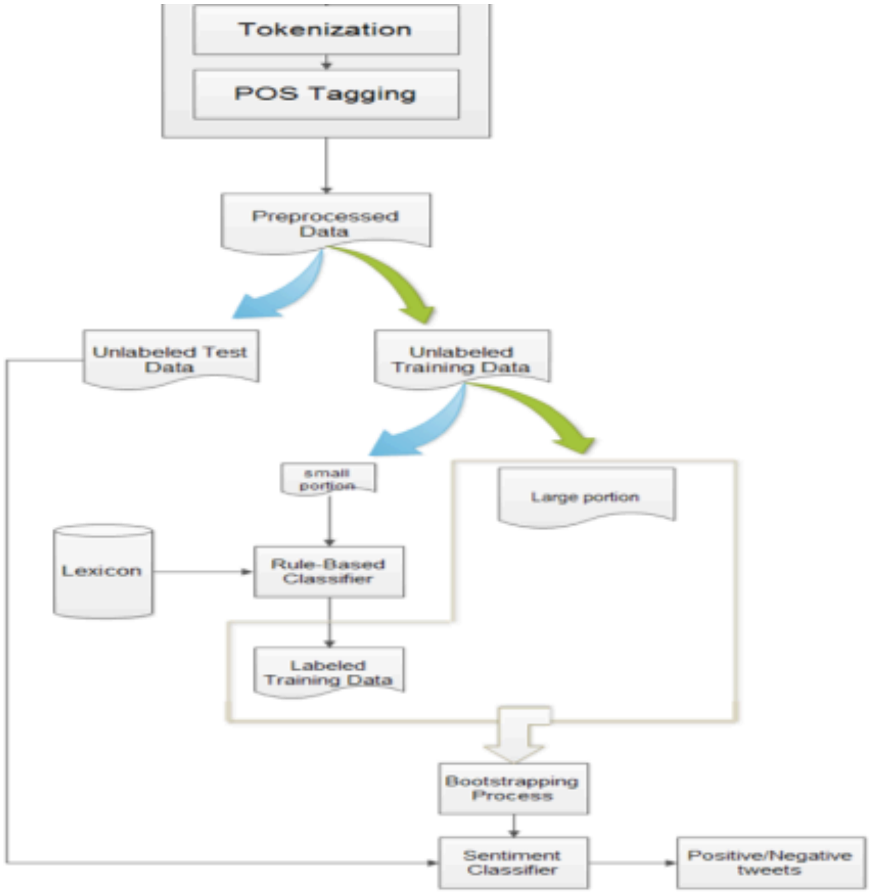
We first constructed an initial word list, containing strong positive and negative sentiment-bearing words, using a Twitter corpus. The word list contained annotated English word with their corresponding POS tags and sentiment indicators as follows: Anger, Anticipation, Disgust, Fear, Joy, Sadness, Surprise, Trust. We combined feature list of these sentiment indicators to provide a sentiment score for each word by giving weights to each sentiment manually. Once we got this lexicon to sentiment mapping, we used the online bilingual dictionary (Ovidhan and Samsad) to perform word by word (POS included) translation to Bengali. This gave us our lexicon list along with it's sentiment polarity/score annotated. Note that the polarity given to the extracted words may not represent the actual sentiment of the words, since the polarity of the extracted words are merely assigned according to their sentence polarity. Hence to verify whether the sentiment label that is given to each extracted word according to its sentence emoticon is actually its polarity, we further apply a filtering method using SentiWordNet. We use SentiWordNet to check each of the translated word for its polarity, according to its POS. In SentiWordNet, a word with a specific POS can have many senses, and each sense is given 3 scores for positivity, negativity, and objectivity, indicating the strength of each of the three properties for that sense of the word. The 3 scores together are equal to 1. As we are dealing with positive and negative sentiment words, we only look at the positive and negative scores and assign the word to the corresponding majority score sentiment.

We use this Bengali polarity lexicon for the rule-based classifier and feature extraction which will be explained in the further sections.

METHODOLOGY

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

Table 1 Architecture



DATASET

The data comes from Indian Language Part-of-Speech Tagset: Bengali, Linguistic Data Consortium (LDC). It is a corpus developed by Microsoft Research (MSR) India to support the task of Part-of-Speech Tagging (POS) and other data-driven linguistic research on Indian Languages in general. This corpus contains 7168 sentences (102933 words) of manually annotated text from modern standard Bengali sources including blogs, Wikipedia, Multikulti and a portion of the EMILLE/CIIL corpus. All annotated data was provided as xml files. Since not all of this data was free for use, we requested a sample of the data containing 372 sentences (4600 words). The sentences we retrieved, include English text as well; instead of filtering out the English texts from the sentences, we included them as part of our training and test sets, since English words express strong positive and negative sentiment.

We also used a dataset of 9000+ words as lexicon to train the sentiment analysis model. This dataset was acquired by the NRC Emotion Lexicon repository. The NRC Emotion Lexicon is a list of English words and their associations with eight basic emotions (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust) and two sentiments (negative and positive). The annotations were manually done by crowdsourcing.

PREPROCESSING

The raw data we obtained through the corpus has some metadata information embedded right into the sentences, and thus need to preprocessed before we proceed with sentiment analysis.

Initially, each sentence is as follows: বা\CCD.n আদৌ\AMN.0.n.n বিয়ে\NC.0.0.n.n করব\VM.3.fut.sim.dcl.fin.n.n.n ।\PU.

Each sentence contains word alongside POS defined as –

Nouns: NC (common noun), NP (proper noun), NV (verbal noun), NST (spatio-temporal noun) Verbs: VM (main verb), VA (auxiliary verb) Pronouns: PPR (pronominal), PRF (reflexive), PRL (relative), PWH (wh-pronoun)Nominal modifiers: JJ (adjective), JQ (quantifier)

Apart from POS information, the data annotates lexical category and types, morphological attributes and their associated values in the context, language, encoding and data size.

We first read the xml formatted data into csv to import as pandas data frame, where we used python to separate all these metadata information from the sentences and store them as a feature to raw sentences. As our dataset contains both Bengali and English sentences, punctuation marks for both are removed. Furthermore, all Bengali punctuations and their usage is similar to that of English (e.g.,'!',';','?',',') ,except for dari ('।'), which is Bengali equivalent of full stop.

Tokenization is performed using custom python function. POS is already part of the dataset, but the information has been used make a POS tagger for other Bengali sentences.

Approach

FEATURES CONSTRUCTION:

Since, we don't have an extensive dataset, we follow the steps of self training bootstrapping. At first we utilize the manually constructed Bengali lexicon, containing strong positive and negative sentiment bearing Bengali words, to label a small number of Bengali sentences based on a certain rule. Since our collected sentences also contain some English words in them, we use the online available English lexicon, which consists of around positive and negative sentiment words.

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

1. If countpositive > countnegative: Label 'positive'

2. If countnegative > countpositive: Label 'negative'

The positive lexicons in our case are the ones associated with the sentiment – Trust, Surprise, Joy and Anticipation; while negative lexicons are associated with sentiments – Anger, Fear and disgust.

That is, words in sentences are compared with the words in the Bengali and Enlgish lexicons for a match. For every match, it is then checked if the word has a positive or a negative label in the polarity lexicons, and count for that label is incremented accordingly. If the count of positive words exceeds that of negative words, the Bengali sentence is labeled as positive and vice versa. Sentences without any lexicon entry or with equal count of positive and negative words are discarded for our experiment.
So, this way our training dataset of annotated sentiments is put together. The final numbers come out to be 211 train sentences and 130 test sentences.

FEATURE EXTRACTION

In feature extraction, each sentence 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 classifier.

The following set of features was used for each sentences:

Word N-gram

We use unigrams and bigrams for our work.

আমার ভাল লাগতাছে না *(I am not feeling well)*

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

Lexicon

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

The presence/absence of each entry in the lexicons is checked per sentence and added as a binary feature in the feature vector for that sentence.

Pos-Tagging

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

CLASSIFIER

fter the training process, we then apply the trained classifier on the test dataset of 300 sentences so that new sentences are labeled as positive or negative. In our work, we use state-of-the-art classifiers for such scenarios, namely, Support Vector Machine.

SVM training algorithm builds a model that assigns new test sentence into one of the two possible classes. The basic idea of the classifier is to find a hyper-plane that separates the positive and negative sentiment classes with maximum margin (or the largest possible distance from both classes). Here, scikit learn has been used as the library to use SVM in python.

Experiment Results and Evaluation

In order to evaluate the performance of our classifier, we first use the standard precision, recall and F- measure to measure the positive and negative sentiments using various sets of features. We then use the accuracy metric to compare the overall performance of the classifier over the baseline classifier of just assigning the majority class. Sentiment analysis task can be interpreted as a classification task where each classification label represents a sentiment. Hence, we define and calculate the four metrics for each label (positive and negative) the same way as in general classification task.

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

Precision (P) = True Positive/(True Positive + False Positive)

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

Recall (R) = True Positive/ (True Positive + False Negative)

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

F-measure = (2*Precision*Recall)/ (Precision + Recall)

RESULTS:

The following are the results obtained on the test set.

Features	**Overall Performance (Average F-score)**
Baseline (Majority Class)	51.4%
All words as features using SVM	48%
SentiWordNet + Lexicon	57%
Lexicon+SentiWordNet+POS	62.1%

As we can infer from the results, the SVM classifier has done a pretty good job performing classification over the baseline classifier. Note that using all words as features actually makes the model perform worse than the baseline since it makes the feature vector extremely sparse. With that in mind, we optimized the features to include POS features that were Noun, Adjectives and Verbs. This dramatically reduced the feature space. Using the optimized features, we created the lexicon list and used it as a feature to create our training data as well as features for the model. All this combined with the POS information led to creation of a robust model that gave a pretty good accuracy on the test dataset.

Reference:

1. http://dspace.bracu.ac.bd/xmlui/bitstream/handle/10361/2902/10101037%20%26%2010101038.pdf?sequence=1&isAllowed=y

2. https://www.academia.edu/16417744/Performing_Sentiment_Analysis_in_Bangla_Microblog_Posts?auto=download

3. 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.
4. Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. Technical report, Stanford.
5. 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.
6. 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

7- http://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm

8- - http://amitavadas.com/Pub/SentiwordNet%20(Bengali).pdf