Sentiment Mining: An approach for Bengali and Tamil Tweets

Sudha Shanker Prasad[1], Jitendra Kumar[2], Dinesh Kumar Prabhakar[3], Sachin Tripathi[4]
Department of Computer Science & Engineering,
Indian Institute of Technology (ISM), Dhanbad, India
[1]sudha.shanker@cse.ism.ac.in, [2]jitendrakumar@cse.ism.ac.in, [3]dinesh.nitr@gmail.com, [4]var_1285@yahoo.com

Abstract—**This paper presents a proposed work for extracting the sentiments from tweets in Indian Language. We proposed a system that deal with the goal to extract the sentiments from Bengali & Tamil tweets. Our aim is to classify a given Bengali or Tamil tweets into three sentiment classes namely positive, negative or neutral. In recent time, Twitter gain much attention to NLP researchers as it is most widely used platform that allows the user to share there opinion in form of tweets. The proposed methodology used unigram and bi-gram models along with different supervised machine learning techniques. We also consider the use of features generated from lexical resources such as Wordnets and Emoticons Tagger.**

Keywords—Sentiment Mining; Information Retrieval; Polarity Identification; Natural Language Processing; Machine learning

I. Introduction

Natural Language Processing is a domain of computer science and scientific study of human language i.e. linguistics which is related with the interaction or interface between the human (natural) language and computer [1]. Sentiment Analysis is one of the major areas of focus in Natural Language Processing (NLP). It deals with opinion mining to determine the sentiment regarding the various topics/subjects in discussion. The basic task is classification of piece of text stating an opinion on an issue with one of the two opposing sentiments (Thumbs up/positive or Thumbs down/negative).

With increase in the number of users in social media, user generated web contents such as chat, blogs, microblogs, etc. are increasing rapidly. This opens a new room for Sentiment Analysis in the social media contents. Twitter is one of the widely used microblogging site. It restricts users to only 140 characters. In addition to this, users use many variations of spellings, various emoticons and improper uses of punctuation marks. These things together made the Sentiment Analysis task more challenging and open new scope for research.

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.

However, during last few years user generated content in other languages is growing at a very rapid rate. Bengali and Tamil are two important Indian languages with a substantially large number of speakers. This paper describes the system we used for sentiment classification of Bengali and Tamil text messages into one of the sentiment classes namely positive, negative or neutral. In order to accomplish the task, we used Naive Bayes and C4.5 Decision Tree algorithm.

Our paper is organized as follows. *Section 2* focuses some of the works already done in this area. *Section 3* describes the system description. In *Section 4,* we discuss our methodology to the problem. *Section 5* reports our classification results. Finally, *Section 6* & *7* deal with discussion and conclusion respectively.

II. Related Work

Sentiment Analysis has been the focus of research community in last decade. In past, several amount of work has been done in this area. Early work in this area includes work done by Turney [2] and Pang [3] for detecting the polarity of product reviews. A multiway document classification on polarity basis is attempted by Pang [4] and Synder [5]. They focuses on classifying movie reviews based on star ratings in either positive or negative. Synder work on restaurant reviews on various aspects like food and ambience. Several subtasks that includes identification of entities, extraction of feature and opinion about the feature either as positive, negative or neutral. More discussion about can be found in Liu's NLP Handbook chapter "Sentiment Analysis and Subjectivity" [6].

In last few years, an inclination of research community Microblogging like Twitter to extract public opinions, to predict the trends of stock markets, outcome of elections [5, 14, 12] and in disaster management[13]. However, very few work in sentiment analysis is done that involves Indian Language that involves work done for Bengali [7]. A strategy for detection of sentiments in Hindi is reported in [8]. A SentiWordNet developed for Bengali using a English-Bengali dictionary and Sentiment Lexicons available in English is proposed by Das and Bandyopadhyay [9]. Labeling emotional expressions in Bengali blog corpus is done by Das and Bandyopadhyay [10]. Deepu and Jisha [11] proposed a rule based approach for sentiment analysis on Malayalam movie review data. A lexicon verb based sentiment classification in Manipuri language is done by Kishorjith and Kumar [12]. Balamurali [13] presents a sentiment classification based on cross lingual that uses feature generated from WordNet. The model is learned from these features using Support Vector Machine.

III. System Description

We describe here the details of our system configuration

978-1-5090-3251-8/16/$31.00 ©2016 IEEE

A. Dataset

The corpus available for training contains 999 Bengali tweets and 1103 Tamil tweets with one of the three annotations namely Positive, Negative and Neutral. The Test Data contains 499 Bengali tweets and 560 Tamil tweets without annotations. We have used WEKA tool in our system for classification purpose.

B. Classifier Used

- *Naive Bayes Classifier:* Naive Bayes classifier is a simple probabilistic classifier based on applying Baye's theorem with strong independence assumptions between the features. Given a problem instance to be classified, represented by a vector $x = (x_1, \dots x_n)$ representing some n features (independent variables). It assigns to this instance probabilities $p(C_k|x_{1,\dots,}x_n)$ for each of K possible outcomes or classes. Using Baye's theorem, the conditional probability can be decomposed as:

$$p(C_k|x) = \frac{p(C_k)\, p(x|C_k)}{p(x)} \qquad (1)$$

where,
$p(C_k)$ = probability of class k
$p(x|C_k)$ = probability of tweet x given class k
$p(x)$ = probability of tweet x

- *C4.5 Decision Tree Classifier:* C4.5 is a classification algorithm based on decision tree approach uses the information gain ratio that is evaluated by entropy. The test feature i.e. the feature selection at each node in the tree is selected using information gain ratio. The attribute with the highest information gain ratio is chosen as the test feature for the current node. Let D be a set consisting of $(D_1, \dots D_n)$ instances. Suppose the class label attribute has m distinct values defining m distinct classes $C_i\ (for\ i = 1, \dots m)$. Let $|D_i|$ be the number of sample of D in class C_i.
The expected information needed to classify a given sample is:

$$Splitinfo_A(D) = -\sum \left(\left.|D_j|\right/_{|D|}\right) * log\left(\left.|Dj|\right/_{|D|}\right)$$

$$Gainratio(A) = Gain(A)/Splitinfo_A(D) \qquad (1)$$

Where,

$$Gain = Info(D) - Info_A(D) \qquad (2)$$

$$Info(D) = -\sum P_i * \log_2(P_i) \qquad (3)$$

And,

$$Info_A(D) = -\sum \left(\left.|D_j|\right/_{|D|}\right) * Info(D_j) \qquad (4)$$

where,
P_i= Probability of distinct class C_i
D = Data Set.
A = Sub-attribute from attribute.
$\left(\left.|D_j|\right/_{|D|}\right)$ = Act as weight of j^{th} partition.

In other words, $gain(A)$ is the expected reduction in entropy caused by knowing the value of feature A.

IV. Proposed Approach

Since the task of Sentiment classification is to predict a sentiment class either positive, negative or neutral for each Bengali and Tamil tweet based on its sentiment, we identified the above task as a three class supervised machine learning classification problem.
Given flowchart describe the proposed approach:

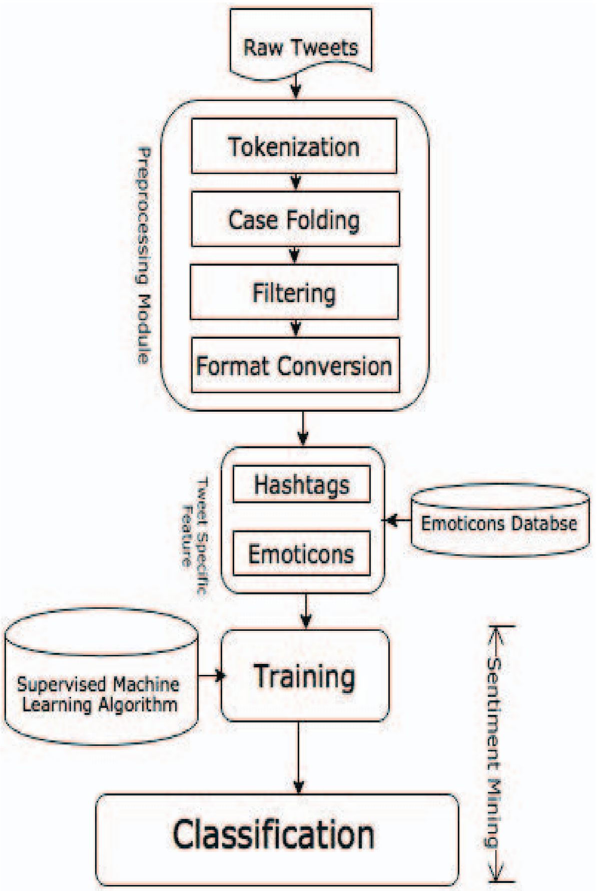

Fig. 1. System Description for Proposed Approach

Following steps will describe the process we have carried out for our proposed work.

A. Preprocessing Module

Following steps are performed for the preprocessing of the tweets both from training and test data.

1) Tokenization: As tweets are user generated content, sometimes user types two or more terms without any white space between them.

2) Case folding: Terms having upper-case letters are converted into its lower-case version.

3) Filtering: Tweets contains URL links, user names and punctuation marks that do not contribute to sentiment and hence need to be removed.

Format Conversion: .arff conversion is carried out.

B. Feature Extraction

This module focuses on identifying features that contributed to the sentiment of the tweets.

1) TF-IDF score of Unigram and Bigram: TF-IDF score gives a statistical measure of an n-gram. This score can be used to identify the inclination of a particular n-gram to any one of the sentiment classes. TF-IDF can be calculated using eq. 1.

$$tf - idf = tf \times \log(N/_{df}) \qquad (1)$$

2) Tweet specific features: Here we considered two features that are specific to tweets namely Hashtags and Emoticons.

- *Hashtags:* Hashtags are the terms starting with a # symbol (for eg. #new_year). It is one of the important features of twitter. We have considered the terms present in a hashtag to find its inclination towards a particular class after removing the # symbol.
- *Emoticons:* Since twitter restricts users to only 140 characters, users frequently uses various emoticons (for eg. ☺) to express their sentiments. We identified this as an important feature and classified the frequently appearing emoticons into the three sentiment classes.

C. Sentiment Classification

The above outlined features are combined with supervised machine learning algorithm to train a classifier. The resulting classifier is used to classify each tweet in the test data into one of the sentiment classes based on its sentiment.

V. Experimental Results

The evaluation of the effectiveness of our system is done with Precision, Recall and F-measure. Table 1 and 2 shows the training accuracy of the model created using Naive Bayes and C4.5 Decision Tree classifier respectively.

TABLE I. Detailed Accuracy Of Bengali Tweets Using Naïve Bayes

Class	Precision	Recall	F_1-score
Negative	0.771	0.819	0.794
Neutral	0.704	0.815	0.755
Positive	0.754	0.399	0.521

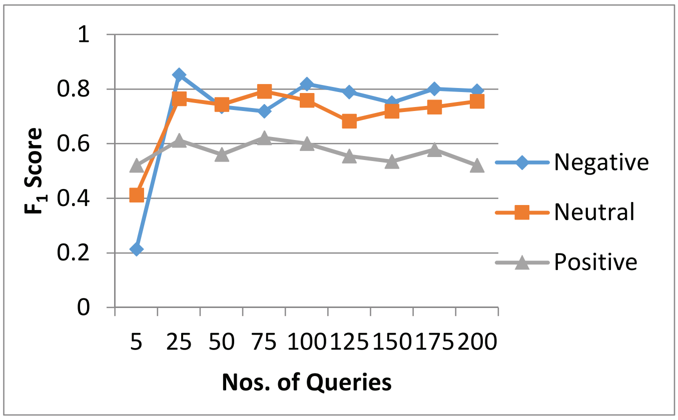

Fig. 1. Bengali Tweets Using Naïve Bayes

TABLE II. Detailed Accuracy Of Bengali Tweets Using C4.5 Decision Tree

Class	Precision	Recall	F_1-score
Negative	0.872	0.889	0.880
Neutral	0.754	0.876	0.810
Positive	0.814	0.391	0.528

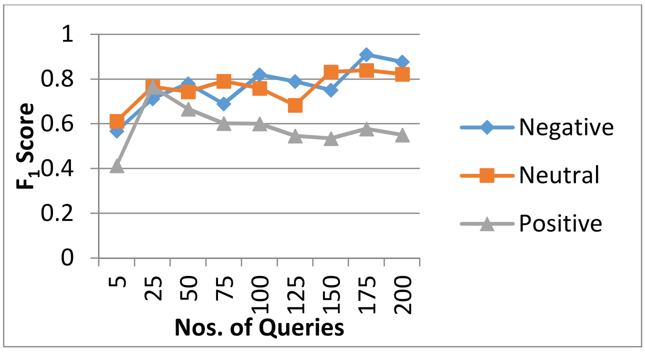

Fig. 2. Bengali Tweets Using C4.5 Decision Tree

TABLE III. Detailed Accuracy Of Tamil Tweets Using Naïve Bayes

Class	Precision	Recall	F_1-score
Negative	0.765	0.812	0.787
Neutral	0.701	0.778	0.737
Positive	0.743	0.394	0.514

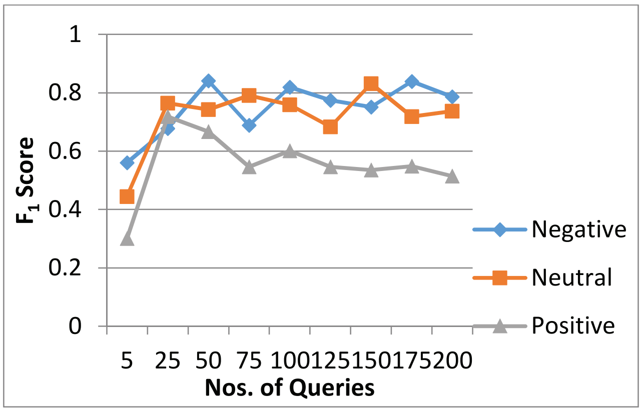

Fig. 3. Tamil Tweets Using Naïve Bayes

TABLE IV. DETAILED ACCURACY OF TAMIL TWEETS USING C4.5 DECISION TREE

Class	Precision	Recall	F_1-score
Negative	0.815	0.842	0.828
Neutral	0.714	0.838	0.771
Positive	0.773	0.373	0.503

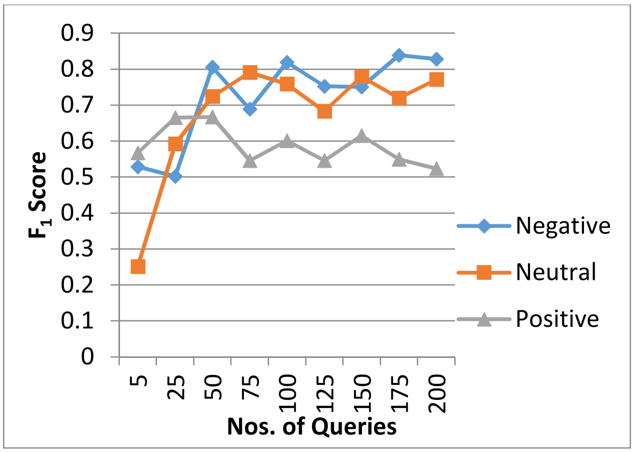

Fig. 4. Tamil Tweets Using C4.5 Decision Tree

VI. DISCUSSION

We found that the task, Sentiment Analysis in Indian Languages is quite challenging due to its nature. One possible reason is the size of the training data. We believe that 999 Bengali and 1103 Tamil tweets are not sufficient enough to build a robust system. As compared to the works already done for English tweets, the accuracy of our system is comparably less. One possible reason for this behavior of our system may be the availability of limited resources in Indian languages. For example, there are some sentiment lexicons available on Indian Languages, but these lexicons are constructed from plain language texts which may not suited for tweets. Because, in case of tweets, there are many variations of spellings (for e.g. "Example"), uses of various emoticons and improper uses of punctuation marks (for e.g. "Example"). In addition to these, tweets in one language also contains terms belonging to other languages (for e.g. "Example") These variations make the sentiment analysis task on Indian languages more difficult and challenging as compared to English.

VII. CONCLUSION

In this paper, we described our experiment on Sentiment Mining in Indian Languages i.e. Bengali & Tamil tweets. The task is to classify tweets in Indian Languages into positive, negative or neutral. For this, we developed systems based on probabilistic and Decision tree algorithms for sentiment classification of Bengali and Tamil tweets. We have proposed an approach to build models using Naive Bayes and C4.5 Decision Tree algorithms for sentiment mining. The above proposed approach can be extended by developing a hybrid model which makes use of POS taggers and language specific features. Improving the accuracy using hybrid model will be our focus of research in future.

REFERENCES

[1] A. Trivedi, A. Srivastava, I. Singh, K. Singh, and S. K. Gupta, "Literature Survey on Design and Implementation of Processing Model for Polarity Identification on Textual Data of English." *IJCSI*, 2011.

[2] P. D. Turney. "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,* pp. 417-424, Association for Computational Linguistics, 2002.

[3] B. Pang, L. Lillian, and S. Vaithyanathan. "Thumbs up?: sentiment classification using machine learning techniques." In *Proceedings of the ACL-02 International Conference on Empirical methods in natural language processing-Vol. 10, Association for Computational Linguistics,* pp. 79-86, 2002.

[4] B. Pang, and L. Lillian, "Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales." In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, Association for Computational Linguistics,* pp. 115-124, 2005.

[5] B. Snyder, and R. Barzilay. "Multiple Aspect Ranking Using the Good Grief Algorithm." In *HLT-NAACL*, pp. 300-307, 2007.

[6] B. Liu. "Sentiment Analysis and Subjectivity." *Handbook of natural language processing*, pp. 627-666, 2010.

[7] A. Das, and S. Bandyopadhyay. "Subjectivity detection in english and bengali: A crf-based approach." In *Proceeding of ICON*, 2009.

[8] S. S. Prasad, J. Kumar, D. K. Prabhakar, and S. Pal. "Sentiment Classification: An Approach for Indian Language Tweets Using Decision Tree." In *International Conference on Mining Intelligence and Knowledge Exploration,* pp. 656-663. Springer International Publishing, 2015.

[9] A. Das, and S. Bandyopadhyay. "SentiWordNet for Bangla.", Knowledge Sharing Event-4: Task 2, 2010.

[10] D. Das, and Sivaji Bandyopadhyay. "Labeling emotion in Bengali blog corpus–a fine grained tagging at sentence level." In *Proceedings of the 8th Workshop on Asian Language Resources,* 2010.

[11] Nair, Deepu S., Jisha P. Jayan, R. R. Rajeev, and Elizabeth Sherly. "SentiMa-sentiment extraction for Malayalam." In *International Conference on Advances in Computing, Communications and Informatics (ICACCI),* pp. 1719-1723, IEEE, 2014.

[12] K. Nongmeikapam, D. Khangembam, W. Hemkumar, S. Khuraijam and S. Bandyopadhyay, "Verb Based Manipuri Sentiment Analysis", *IJNLC*, vol. 3, no. 3, pp. 113-119, 2014.

[13] A. R. Balamurali, "Cross-lingual sentiment analysis for Indian languages using linked wordnets.", 2012.