International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 05 Issue: 08 | Aug - 2021 ISSN: 2582-3930

Sentiment Analysis for Depression Based On Social Media Post

Poorva Joshi, Prajakta Lawand

Department of Computer Engineering, Pillai HOC College of Engineering and Technology, Rasayani, Maharashtra, India

Swapnil Waghmare, Assistant Professor

Department of Computer Engineering, Pillai HOC College of Engineering and Technology, Rasayani, Maharashtra, India

Abstract— This project addresses the problem of sentiment analysis in twitter; that is classifying tweets according to the sentiment expressed in them; positive, negative or neutral. Twitter is an online micro-blogging and social-networking platform which allows users to write short status updates of maximum length 140 characters. It is a rapidly expanding service with over 200 million registered users – out of which 100 million are active users and half of them log on twitter on a daily basis – generating nearly 250 million tweets per day, Due to this large amount of usage we hope to achieve a reflection of public sentiment by analyzing the sentiments expressed in the tweets. Analysing the public sentiment is important for many applications such as firms trying to find out the response of their products in the market, predicting political elections and predicting social economics phenomena like stock exchange. The aim of this project is to develop a functional classifier for accurate and automatic sentiment classification of an unknown tweet stream.

Keywords—Twitter, Sentiment analysis (SA), Opinion mining, Machine learning, Naive Bayes (NB), Maximum Entropy, Support Vector Machine (SVM).., Natural Language Toolkit¶ (NLTK), Natural Language Processing(NLP)

1 INTRODUCTION

1.1 MOTIVATION

We have chosen to work with twitter since we feel it is a better approximation of public sentiment as opposed to conventional internet articles and web blogs. The reason is that the amount of relevant data is much larger for twitter, as compared to traditional blogging sites. Moreover the response on twitter is more prompt and also more general (since the number of users who tweet is substantially more than those who write web blogs on a daily basis).

Sentiment analysis of public is highly critical in macro-scale socioeconomic phenomena like predicting the stock market rate of a particular firm. This could be done by analyzing overall public sentiment towards that firm with respect to time and using economics tools for finding the correlation between public sentiment and the firm's stock market value. Firms can also estimate how well their product is responding in the market, which areas of the market is it having a favourable response and in which a negative response (since twitter allows us to download stream of geo-tagged tweets for particular locations. If firms can get this information they can analyze the reasons behind geographically differentiated response, and so they can market their product in a more optimized manner by looking for appropriate solutions like creating suitable market segments.

1.2 DOMAIN INTRODUCTION

This project of analyzing sentiments of tweets comes under the domain of "Pattern Classification" and "Data Mining". Both of these terms are very closely related and intertwined, and they can be formally defined as the process of discovering "useful" patterns in large set of data, either automatically (unsupervised) or semiautomatically (Supervised). The project would heavily rely on techniques of "Natural Language Processing in extracting significant patterns and features from the large data set of tweets and on "Machine Learning techniques for accurately classifying individual unlabelled data samples (tweets) according to whichever pattern model best describes them.

2 LITERATURE SURVEY

2.1 SENTIMENT ANALYSIS

Sentiment analysis can be defined as a process that automates mining of attitudes, opinions, views and emotions from text, speech, tweets and database sources through Natural Language Processing (NLP). Sentiment analysis involves classifying opinions in text into categories like "positive" or

"negative" or "neutral". It's also referred as subjectivity analysis, opinion mining, and appraisal extraction. The words opinion, sentiment, view and belief are used interchangeably but there are differences between them.

- View: subjective opinion
- Belief: deliberate acceptance and intellectual assent
- Sentiment: opinion representing one's feelings

An example for terminologies for Sentiment Analysis is as given below, = The story of the movie was weak and boring = Sentiment Analysis is a term that include many tasks such as sentiment extraction, sentiment classification, subjectivity classification, summarization of opinions or opinion spam detection, among others. It aims to analyze people's sentiments, , attitudes, opinions emotions, etc. towards elements such as, products, individuals, topics ,organizations, and services. Mathematically we can represent an opinion as a quintuple (o, f, so, h, t), where o = object; f = feature of the object o; so= orientation or polarity of the opinion on feature f of object o; h = opinion holder; t = time when the opinion is expressed. Object: An entity which can be a, person, event, product ,organization, or topic Feature: An attribute (or a part) of the object with respect to which evaluation is made. Opinion orientation or polarity: The orientation of an opinion on a feature f represent whether the opinion is positive, negative or neutral . Opinion holder: The holder of an opinion is the person or organization or an entity that expresses the opinion . In recent years a lot of work has been done in the field of "Sentiment Analysis on Twitter" by number of researchers. In its early stage it was intended for binary classification which assigns opinions or reviews to bipolar classes such as positive or negative only. Pak and Paroubek (2010) [1] proposed a model to classify the tweets as objective, positive and negative. They created a twitter corpus by collecting tweets using Twitter API and automatically annotating those tweets using emoticons. Using that corpus, they developed a sentiment classifier based on the multinomial Naive Bayes method that uses features like Ngram and POS-tags. The training set they used was less efficient since it contains only tweets having emoticons. Parikh and Movassate(2009) [2] implemented two models, a Naive Bayes bigram model and a Maximum Entropy model to classify tweets. They found that the Naive Bayes classifiers worked much better than the Maximum Entropy model. Go and L.Huang (2009) [3] proposed a solution for sentiment analysis for twitter data by using distant supervision, in which their training data consisted of tweets with emoticons which served as noisy labels. They build models using Naive Bayes, MaxEnt and Support Vector Machines (SVM). Their feature space consisted of unigrams, bigrams and POS. They concluded that SVM outperformed other models and that unigram were more effective as features. Barbosa et al.(2010) [4] designed a two phase automatic sentiment analysis

- Opinion: A conclusion open to dispute (because different experts have different opinions)

method for classifying tweets. They classified tweets as objective or subjective and then in second phase, the subjective tweets were classified as positive or negative. The feature space used included retweets, hashtags, link, punctuation and exclamation marks in conjunction with features like prior polarity of words and POS. Bifet and Frank(2010) [5] used Twitter streaming data provided by Firehouse API , which gave all messages from every user which are publicly available in real-time. They experimented multinomial naive Bayes, stochastic gradient descent, and the Hoeffding tree. They arrived at a conclusion that SGD-based model, when used with an appropriate learning rate was the better than the rest used. Agarwal et al. (2011)[6] developed a 3-way model for classifying sentiment into positive, negative and neutral classes. They experimented with models such as: unigram model, a feature based model and a tree kernel based model. For tree kernel based model they represented tweets as a tree. The feature based model uses 100 features and the unigram model uses over 10,000 features. They arrived on a conclusion that features which combine prior polarity of words with their parts-of-speech(pos) tags are most important and plays a major rolein the classification task. The tree kernel based model outperformed the other two models. Davidov et al.,(2010) [7] proposed a approach to utilize Twitter user-defined hastags in tweets as a classification of sentiment type using punctuation, single words, n-grams and patterns as different feature types, which are then combined into a single feature vector for sentiment classification. They made use of K-Nearest Neighbor strategy to assign sentiment labels by constructing a feature vector for each example in the training and test set. Po-Wei Liang et.al.(2014) [8] used Twitter API to collect twitter data. Their training data falls in three different categories (camera, movie , mobile). The data is labeled as positive, negative and nonopinions. Tweets containing opinions were filtered. Unigram Naive Bayes model was implemented and the Naive Bayes simplifying independence assumption was employed. They also eliminated useless features by using the Mutual Information and Chi square feature extraction method. Finally , the orientation of an tweet is predicted. i.e. positive or negative. Pablo et. al. [9] presented variations of Naive Bayes classifiers for detecting polarity of English tweets. Two different variants of Naive Bayes classifiers were built namely Baseline (trained to classify tweets as positive, negative and neutral), and Binary (makes use of a polarity lexicon and classifies as positive and negative. Neutral tweets neglected). The features considered by classifiers were Lemmas (nouns, verbs, adjectives and adverbs), Polarity Lexicons, and Multiword from different sources and Valence

Shifters. Turney et al [11] used bag-of-words method for sentiment analysis in which the relationships between words was not at all considered and a document is represented as just a collection of words. To determine the sentiment for the whole document, sentiments of every word was determined and those values are united with some aggregation functions. Kamps et al. [12] used the lexical database WordNet to determine the emotional content of a word along different dimensions. They developed a distance metric on WordNet and determined semantic polarity of adjectives. Xia et al. [13] used an ensemble framework for Sentiment Classification which is obtained by combining various feature sets and classification techniques. In their work, they used two types of feature sets (Part-of-speech information and Word relations) and three base classifiers (Naive Bayes, Maximum Entropy and Support Vector Machines) . They applied ensemble approaches like fixed combination, weighted combination and Meta-classifier combination for sentiment classification and obtained better accuracy. Luoet. al. [14] highlighted the challenges and an efficient techniques to mine opinions from Twitter tweets. Spam and wildly varying language makes opinion retrieval within Twitter challenging task.

2.2 Pre-processing of the datasets

A tweet contains a lot of opinions about the data which are expressed in different ways by different users .The twitter dataset used in this survey work is already labeled into two classes viz. negative and positive polarity and thus the sentiment analysis of the data becomes easy to observe the effect of various features. The raw data having polarity is highly susceptible to inconsistency and redundancy. Preprocessing of tweet include following points,

- Remove all URLs (e.g. www.xyz.com), hash tags (e.g. #topic), targets (@username)
- Correct the spellings; sequence of repeated characters is to be handled
- Replace all the emoticons with their sentiment.
- Remove all punctuations ,symbols, numbers
- Remove Stop Words
- Expand Acronyms(we can use a acronym dictionary)
- Remove Non-English Tweets

3 RELATED WORK

Machine learning and deep learning is useful for various set of problems. One of the applications of this technique is in predicting a dependent variable from the values of independent variables. Machine Learning is a part of Artificial Intelligence. ML finds a solution to the problems by recognizing patterns in the databases rather than depending on rules. The Machine Learning techniques such as linear regression and naive Bayes methods study the correlation between features and the value of the output class. In other words, ML techniques help machines to understand some information about the real world [2].

4 PROPOSED WORK

Problem statement: The problem at hand consists of two subtasks: Phrase Level Sentiment Analysis in Social Media (Twitter) Give message containing a marked instance of a word or a phrase, determine whether that instance is positive, negative or neutral in that context. Sentence Level Sentiment Analysis on Social Media (Twitter): Given message, decide whether the message is of positive, negative, or neutral sentiment. For messages conveying both a positive and negative sentiment, whichever is the stronger sentiment should be chosen.

4.1 Methodology

There are 5 steps to analyze sentiment data and here's the graphical representation of the methodology to do the same

Fig 1

Σ A thorough study of existing approaches and techniques in field of sentiment analysis.

Σ Collection of related data from Twitter with the help of Twitter API

Σ Pre-processing of data collected from Twitter so that it can be fit for mining.

Σ To build a classifier based on different supervised machine learning techniques.

Σ Training and testing of build classifier using large datasets

Σ Computing the result of different classifier using dataset collected from Twitter.

Comparing results of each classifier and plotting a graph that show the trend of positive and negative sentiment for different users.

5 SYSTEM IMPLEMENTATION

Data collection is not a simple task, as it may seem. Various decisions have to be made for collecting data. For our thesis we maintain dataset training, testing and for twitter sentiment analysis. In this chapter we are going to study how data is collected, stored, processed and classified. Before discussing these process and different dataset, let us discuss our proposed architecture.

5.1 Proposed Architecture

As our goal is to achieve sentiment analysis for data provided from Twitter. We are going to build a classifier which consists of different machine learning classifiers. Once our classifier is ready and trained we are going to follow the steps shown in

SYSTEM ARCHITECTURE/DATA MODEL

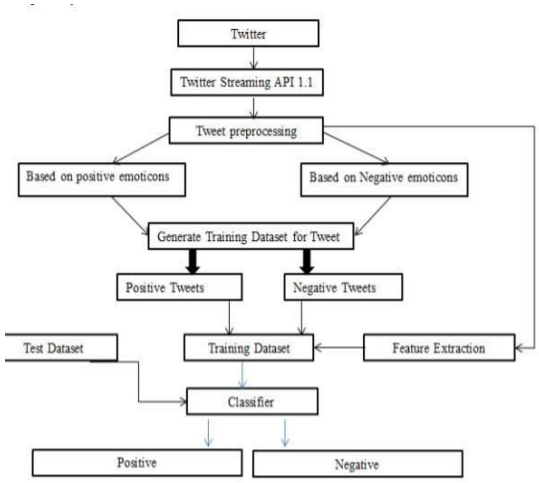

Fig 2

Step-1 First we are going to stream tweets in our build classifier with the help of Tweepy library in python

Step-2 Then we pre-process these tweets, so that they can be fit for mining and feature extraction.

Step-3 After pre-processing we pass this data in our trained classifier, which then classify them into positive or negative class based on trained results.

Since, Twitter is our source of data for analysis. We are going to stream the tweets from twitter in our database. For this we are going to use Twitter Application.

5.2 Twitter API (Application Programming Interface)

Twitter allows users to collect tweets with the help of Twitter API. Twitter provides two kinds of APIS: REST API and Streaming API. The differences between these are: REST APIs support connections for short time interval and only limited data can be collected at a time, whereas Streaming API provides tweets in real-time and connection for long time. We use Streaming API for our analysis. For collecting large amount of tweets we need long-lived connection and no limit data rate.

5.3 Data Collection

5.3.1 Twitter Data

To use Twitter API we must first have a twitter account. It can be easily created by filling the sign up details in twitter.com website. After this you will be provided with a username and password which is use for login purpose. Once your account is created, you can now read and send tweets on any topic you want to explore.

Twitter provider a platform from which we can access data from twitter account and can use it for our own purpose. For this we have to login with our twitter credentials in dev.twitter.com website. In this website, we first create an application which will be used for streaming tweets by providing necessary details. Once our API is created we can get to know customer key, customer secret key, access token key and access secret key. These keys are used to authenticate user when user want to access twitter data.

As the objective of this thesis is to analyze the sentiment of Tweets posed for political parties, only tweets about related to this should be collected. Hence for this we create a Python script which will be used to fetch tweets from twitter. Before creating this script we first install a library in Python called tweepy.

Python is a very powerful language which provides many services with the help of many Python libraries. Tweepy is one of the open source Python library which enables Python to communicate with twitter and use its API to collect data so that we can use it in our program. To install tweepy, just provide a command 'pip install tweepy' in command prompt or bash and we ready to go with script.

In this script we use all the keys and secrets which we got in API, we first create listener class which is used to load the data from the twitter. Now to gather data we first set up OAuth' protocol. OAuth is a standard protocol which is used for authorization. It allow user to log in any third party websites by using any social network website account without exposing passwords. OAuth provides security and authorization to user.

The script which we use to access data with the help of twitter is shown below

```
from tweepy import Stream
from tweepy import OAuthHandler
from tweepy.streaming import StreamListener

#consumer key, consumer secret, access token, access secret.
ckey="####"
csecret="####"
atoken="####"
asecret="####"

class listener(StreamListener):

    def on_data(self, data):
        print(data)
        saveFile = open('twitDB.csv','a')
        saveFile.write(data)
        saveFile.write('\n')
        saveFile.close()
        return(True)

    def on_error(self, status):
        print(status)

auth = OAuthHandler(ckey, csecret)
auth.set_access_token(atoken, asecret)

twitterStream = Stream(auth, listener())
twitterStream.filter(track=["    "])
```

In this script we have to provide all the keys which are given by Twitter API. To get the tweet for a particular topic we import Stream' library from tweepy. In this we pass the authorization detail and the class in which we import tweets. We also apply a filter in the stream which will help us to provide the tweets for the particular topic by providing a keyword related to that topic in filter. Once we run our script, we see tweets are imported from Twitter and we can then use them for our purpose.

5.3.2 Training Data

Other data which we collected for this thesis is training data. This data is used to train the classifier which we are going to build. To collect this data we use NLTK library of Python. NLTK consists of corpora, which is very large and consists of structured set of text files which are used to perform analysis. In these corpora there are various types of text files like quotes, reviews, chat, history, etc. From these corpora we will select files of movie reviews for our training purpose. Sample of these reviews is shown in Table

Movie Reviews	CLASS
foolish, idiotic and boring it's so lad dish and youngish , only teenagers could find it funny	NEGATIVE
the rock is destined to be the 21st century's new conan and that he's going to make a splash even greater than arnold schwarzenegger	POSITIVE
Barry Sonnenfeld owes frank the pug big time the biggest problem with roger avary's uproar against the map	NEGATIVE
the seaside splendor and shallow , beautiful people are nice to look at while you wait for the story to get going	POSITIVE

Table 1

In reviews corpus there are around 5000 reviews each for positive and negative feedback. These reviews are short and arranged in text files which are easy to access. We train our classifier from around 80% of the data and then we test it with remaining 20% to check that trained classifier is working properly or not.

5.4 Data Storage

Once, we start getting our data from Twitter API our next step is to store that data so that we can use it for sentiment analysis. We ran our scripts for period of month and collect the tweets for different political parties. Every time we ran the script described in figure a .csv (comma separated values) file is generated which consists of tweets that are extracted from Twitter API. We use .csv format for our collected data files because data consists of many fields. CSV separate each field with a comma, thus make it very easier to access the particular field which consists of text. CSV files also provide faster read/write time as compared to others.

We make separate directories to store tweets of different political parties for respective month. We store them in our hard drive from where these can be easily imported to our snippet and further proceed for analysis. Once we stored our tweet we have to pre process the data stored before applying it to classifier because the data we collect from API is not fit for mining. Therefore pre-processing the data is our next step.

5.5 Data Pre-Processing

Data obtained from twitter is not fit for extracting features. Mostly tweets consists of message along with usernames, empty spaces, special characters, stop words, emoticons, abbreviations, hash tags, time stamps, URL's ,etc. Thus to make this data fit for mining we pre-process this data by using various function of NLTK. In pre processing we first extract our main message from the tweet, then we remove all empty spaces, stop words (like is, a, the, he, them, etc.), hash tags, repeating words, URL's, etc. We then replace all emoticons and abbreviations with their corresponding meanings like :-), D. =), LOL, Rolf, etc. are replaced with happy or laugh. Once

we are done with it, we are ready with processed tweet which is provided to classifier for required results.

Cleaning of Twitter data is necessary, since tweets contain several syntactic features that may not be useful for analysis. The pre-processing is done in such a way that data represented only in terms of words that can easily classify the class.
We create a code in Python in which we define a function which will be used to obtain processed tweet. This code is used achieve the following functions:

Σ remove quotes - provides the user to remove quotes from the text

Σ remove @- provides choice of removing the @symbol, removing the @ along with the user name, or replace the @ and the user name with a word 'AT_USER' and add it to stop words

Σ remove URL (Uniform resource locator) - provides choices of removing URLs or replacing them with ' URL' word and add it to stop words

Σ remove RT (Re-Tweet) - removes the word RT from tweets

Σ remove Emoticons remove emoticons from tweets and replace them with their specific meaning

Σ remove duplicates - remove all repeating words from text so that there will be no duplicates

Σ remove # - removes the hash tag class Σ remove stop words remove all stop words like a, he, the, and, etc which provides no meaning for classification.

It shows the various types of contents that are included in tweets and also the actions performed on these contents.
Once our data is cleaned and ready for processing our next step is to classify this cleaned data into different classes. For this we have to use supervise machine learning classifiers.

5.6 Classification

To classify tweets in different class (positive and negative) we build a classifier which consists of several machine learning classifiers. To build our classifier we used a library of Python called, Scikit-learn. Scikit-learn is a very powerful and most useful library in Python which provides many classification algorithms. Scikit-learn also include tools for classification, clustering, regression and visualization. To install Scikit-learn we simply use on line command in python which is pip install scikit learn'.

In order to build our classifier, we use seven in-build classifiers which come in Scikit learn library, which are:

Σ Naïve-Bayes Classifier
Σ MultinomialNB Classifier
Σ BernoulliNB Classifier
Σ Logistic Regression Classifier
Σ SGDC
Σ Linear SVC
Σ Nu SVC

The reason we are using seven classifiers, so that we can get the more reliable output. To use these classifiers, we write a script in Python, in which we first import the classifier and then we pass the training set to each classifier.

5.6.1 Feature Extraction

As we already discussed training and testing data is collected from NLTK corpus. We have round 5000 movie reviews each for positive and negative class. We take first 4000 reviews as training set and remaining 1000 as testing sets.

Both the training and testing data must be represented in same order for learning. One of the ways that data can be represented is feature-based. By features, it is meant that some attributes that are thought to capture the pattern of the data are first selected and the entire dataset must be represented in terms of them before it is fed to a machine learning algorithm. Different features such as n-gram presence or n-gram frequency, POS (Part of Speech) tags, syntactic features, or semantic features can be used. For example, one can use the keyword lexicons as features. Then the dataset can be represented by these features using either their presence or frequency.

Attribute selection is the process of extracting features by which the data will be represented before any machine learning training takes place. Attribute selection is the first task when one intends to represent instances for machine learning. Once the attributes are selected, the data will be represented using the attributes. So attributes are the features. Although we used the entire data set in our selection of attributes, the representation of the data must be done on a per instance (Twitter post) basis.

Feature vector plays a very important role in classification and helps to determine the working of the build classifier. Feature vector also help in predicting the unknown data sample. There are many types of feature vectors, but in this process we used unigram approach. Each tweet words are added to generate the feature vectors. The presence/absence of sentimental word

Existing System	Proposed System
Existing system takes a stored dataset on a particular topic into consideration.	Proposed system will gives you the freedom to choose the data of any topic.
It fails to determine the impact the results might or will have in the respective field.	Here, it gives you the impact the results and statistics will have on the respective field.
Existing system does not allow the retrieval of data based on the query entered by user.	Proposed system allows retrieval of data based on the query entered by the user.
Existing system does not provide accurate feature selection.	Proposed system will provide accurate feature selection.

helps to indicate the polarity of the sentences. We create a python script to extract the features from the training data. Code snippet for extracting features is shown in Figure 4.4 Once we extract the features from training data, we are going to pass these in our build classifiers. A script in written in python which is used to pass training sets in classifier. Once, the classifier is trained we can also check the accuracy of each classifier by passing the testing set.

6. RESULTS

We used the twitter dataset publicly made available by Stanford university. Analyses was done on this labelled datasets using various feature extraction technique. We used the framework where the pre-processor is applied to the raw sentences which make it more appropriate to understand. Further, the different machine learning techniques trains the dataset with feature vectors and then the semantic analysis offers a large set of synonyms and similarity which provides the polarity of the content.

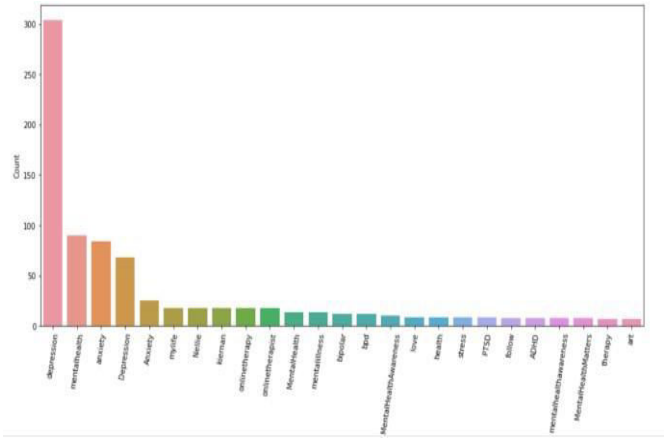

Fig 3

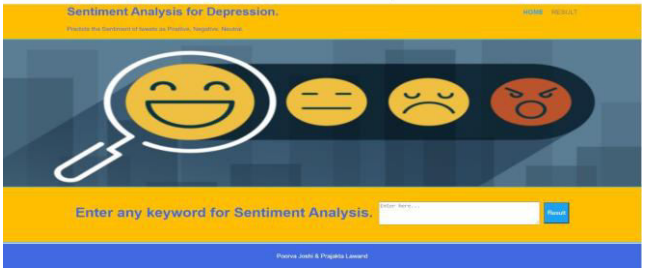

Fig 4

Fig 5

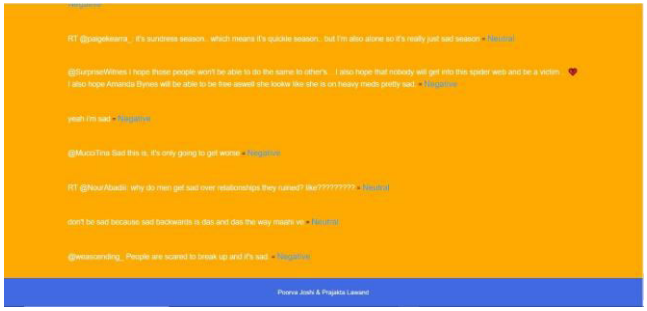
Fig 6

Comparison between existing system and proposed system

Table 2

7 APPLICATIONS OF SENTIMENT ANALYSIS

Sentiment Analysis has many applications in various Fields.

1.Applications that use Reviews from Websites: Today Internet has a large collection of reviews and feedbacks on almost everything. This includes product reviews, feedbacks on political issues, comments about services, etc. Thus there is a need for a sentiment analysis system that can extract sentiments about a particular product or services. It will help us to automate in provision of feedback or rating for the given product, item, etc. This would serve the needs of both the users and the vendors. 2. Applications as a Sub-component Technology A sentiment predictor system can be helpful in recommender systems as well. The recommender system will not recommend items that receive a lot of negative feedback or fewer ratings. In online communication, we come across abusive language and other negative elements. These can be detected simply by identifying a highly negative sentiment and correspondingly taking action against it.

3. Applications in Business Intelligence It has been observed that people nowadays tend to look upon reviews of products which are available online before they buy them. And for many businesses, the online opinion decides the success or failure of their product. Thus,

Sentiment Analysis plays an important role in businesses. Businesses also wish to extract sentiment from the online reviews in order to improve their products and in turn their reputation and help in customer satisfaction .

4. Applications across Domains: Recentre searches in sociology and other fields like medical, sports have also been benefitted by Sentiment Analysis that show trends in human emotions especially on social media.

8 CONCLUSION

In this paper, we provide a survey and comparative study of existing techniques for opinion mining including machine learning and lexicon-based approaches, together with cross domain and cross-lingual methods and some evaluation metrics. Research results show that machine learning methods, such as SVM and naive Bayes have the highest accuracy and can be regarded as the baseline learning methods, while lexicon-based methods are very effective in some cases, which require few effort in human-labeled document .We also studied the effects of various features on classifier. We can conclude that more the cleaner data, more accurate results can be obtained. Use of bigram model provides better sentiment accuracy as compared to other models. We can focus on the study of combining machine learning method into opinion lexicon method in order to improve the accuracy of sentiment classification and adaptive capacity to variety of domains and different languages.

9 REFERENCES

[1] A.Pak and P. Paroubek. „Twitter as a Corpus for Sentiment Analysis and Opinion Mining". In Proceedings of the Seventh Conference on International Language Resources and Evaluation, 2010, pp.1320-1326

[2] R. Parikh and M. Movassate, “Sentiment Analysis of User- Generated Twitter Updates using Various Classi_cation Techniques",CS224N Final Report, 2009 [3] Go, R. Bhayani, L.Huang. “Twitter Sentiment Classification Using Distant Supervision". Stanford University, Technical Paper,2009

[4] L. Barbosa, J. Feng. “Robust Sentiment Detection on Twitter from Biased and Noisy Data". COLING 2010: Poster Volume, pp. 36-44.

[5] Bifet and E. Frank, "Sentiment Knowledge Discovery in Twitter Streaming Data", In Proceedings of the 13th International Conference on Discovery Science, Berlin, Germany: Springer,2010, pp. 1-15.

[6] Agarwal, B. Xie, I. Vovsha, O. Rambow, R. Passonneau, “Sentiment Analysis of Twitter Data", In Proceedings of the ACL 2011Workshop on Languages in Social Media,2011 , pp. 30-38

[7] Dmitry Davidov, Ari Rappoport." Enhanced Sentiment Learning Using Twitter Hashtags and Smileys". Coling 2010: Poster Volume pages 241{249, Beijing, August 2010

[8] Po-Wei Liang, Bi-Ru Dai, “Opinion Mining on Social Media Data", IEEE 14th International Conference on Mobile Data Management ,Milan, Italy, June 3 - 6, 2013, pp 91-96, ISBN:978-1-494673-6068-5, http://doi.ieeecomputersociety.org/10.1109/MDM.2013.

[9] Pablo Gamallo, Marcos Garcia, “Citius: A Naive-Bayes Strategy for Sentiment Analysis on English Tweets", 8th International Workshop on Semantic Evaluation (Sem Eval 2014), Dublin, Ireland, Aug 23-24 2014, pp 171-175.

[10] Neethu M,S and Rajashree R,” Sentiment Analysis in Twitter using Machine Learning Techniques”

4th ICCCNT 2013,at Tiruchengode, India. IEEE – 31661
[11] 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.

[12] J. Kamps, M. Marx, R. J. Mokken, and M. De Rijke, "Using wordnet to measure semantic orientations of adjectives," 2004.

[13] R. Xia, C. Zong, and S. Li, "Ensemble of feature sets and classification algorithms for sentiment classification," Information Sciences: an International Journal, vol. 181, no. 6, pp. 1138–1152, 2011.

[14] ZhunchenLuo, Miles Osborne, TingWang, An effective approach to tweets opinion retrieval", Springer Journal on World Wide Web ,Dec 2013, DOI: 10.1007/s11280-013-0268-7.