Volume 5, Issue 3, March 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Sentiment Analysis-Time Variant Analytics

D. Sai Krishna, G **Akshay Kulkarni, A. Mohan**
CSE Department, CBIT
Telangana, India

Abstract: Microblogging websites have evolved to become a source of varied kind of information. This is due to nature of micro blogs on which people post real time messages about their opinions on a variety of topics, discuss current issues, complain, and express positive sentiment for products they use in daily life. In fact, companies manufacturing such products have started to poll these microblogs to get a sense of general sentiment for their product. Many times these companies study user reactions and reply to users on microblogs. One challenge is to build technology to detect and summarize an overall sentiment. In this paper, we look at one such popular micro blog Called Twitter and build models for classifying "tweets" into positive, negative and neutral sentiment. We perform Real time sentiment analysis on twitter tweets.

Key words: sentiment, opinion, real time, twitter.

I. INTRODUCTION

Sentimental Analysis is all about to get the real voice of people towards specific product, services, organization, movies, news, events, issues and their attributes. Sentiment Analysis includes branches of computer science like Natural Language Processing, Machine Learning, Text Mining and Information Theory and Coding. By using approaches, methods, techniques and models of defined branches, we can categorize our data which is unstructured data which may be in form of news articles, blogs, tweets, movie reviews, product reviews etc. into positive, negative or neutral sentiment according to the sentiment is expressed in them. It is used for tracking the mood of the public about a particular product or topic. Sentiment analysis can be useful in several ways. For example, in marketing it helps in judging the success of an ad campaign or new product launch, determine which versions of a product or service are popular and even identify which demographics like or dislike particular features.

There are several challenges in Sentiment analysis. The first is a opinion word that is considered to be positive in one situation may be considered negative in another situation. A second challenge is that people don't always express opinions in a same way. Most traditional text processing relies on the fact that small differences between two pieces of text don't change the meaning very much. In Sentiment analysis, however, "the picture was great" is very different from "the picture was not great". People can be contradictory in their statements. Most reviews will have both positive and negative comments, which is somewhat manageable by analysing sentences one at a time. However, in the more informal medium like twitter or blogs, the more likely people are to combine different opinions in the same sentence which is easy for a human to understand, but more difficult for a computer to parse. Sometimes even other people have difficulty understanding what someone thought based on a short piece of text because it lacks context. [20]

Sentiment analysis is done on three levels [20]

- Document Level
- Sentence Level
- Entity or Aspect Level

Document Level Sentiment analysis is performed for the whole document and then decide whether the document express positive or negative sentiment. [3][20]

Entity or Aspect Level sentiment analysis performs finer-grained analysis. The goal of entity or aspect level sentiment analysis is to find sentiment on entities and/or aspect of those entities. For example consider a statement "My HTC Wildfire S phone has good picture quality but it has low phone memory storage." so sentiment on HTC's camera and display quality is positive but the sentiment on its phone memory storage is negative. [3]

Sentence level sentiment analysis is related to find sentiment form sentences whether each sentence expressed a positive, negative or neutral sentiment. Sentence level sentiment analysis is closely related to subjectivity classification. Many of the statements about entities are factual in nature and yet they still carry sentiment. We can generate summery of opinions about entities. Comparative statements are also part of the entity or aspect level sentiment analysis but deal with techniques of comparative sentiment analysis. This paper deals with the sentence and entity level sentiment analysis and classifies the real time twitter tweets into positive, negative and neutral tweets using naïve Bayes' classifier.

Languages that have been studied mostly are English and in Chinese .Presently, there are very few researches conducted on sentiment classification for other languages like Arabic, Italian and Thai. This survey aims at focusing much of the work in English. The emergence of sentiment analysis dates back to late 1990's, but becomes a major

emerging sub field of information management discipline only from 2000, especially from 2004 onwards, which this paper focuses.

The rest of the paper is organized as follows. In section 2, we discuss data sources on which sentiment analysis can be performed. In section 3, we focus on how to TwitterAPI is used to extract real time data(tweets) from Twitter. In section 4 Naïve Bayes' classification technique used for classifying tweets into positive, negative and neutral sentiments. In section 5 we present timestamp based scoring scheme. In section 6 we present the design of our web app. In section 7 we present our experiments and discuss the results. We conclude and give future directions of our work in section 8.

II. DATA SOURCE

Users' opinion is a major criterion for the improvement of the quality of services rendered and enhancement of the deliverables. Blogs, review sites, data and micro blogs provide a good understanding of the reception level of the products and services.[1]

2.1. Blogs

With an increasing usage of the internet, blogging and blog pages are growing rapidly. Blog pages have become the most popular means to express one's personal opinions. Bloggers record the daily events in their lives and express their opinions, feelings, and emotions in a blog. Many of these blogs contain reviews on many products, issues, etc.[15] Blogs are used as a source of opinion in many of the studies related to sentiment analysis.[16]

2.2. Review sites

For any user in making a purchasing decision, the opinions of others can be an important factor. A large and growing body of user-generated reviews is available on the Internet. The reviews for products or services are usually based on opinions expressed in much unstructured format. The reviewer's data used in most of the sentiment classification studies are collected from the e-commerce websites like www.amazon.com (product reviews), www.yelp.com (restaurant reviews), www.CNET download.com (product reviews) and www.reviewcentre.com, which hosts millions of product reviews by consumers. Other than these the available are professional review sites such as www.dpreview.com, www.zdnet.com and consumer opinion sites on broad topics and products such as www .consumerreview.com.[17][18]

2.3. Micro-blogging

Twitter is a popular microblogging service where users create status messages called "tweets". These tweets sometimes express opinions about different topics. Twitter messages are also used as data source for classifying sentiment.

We have chosen Twitter as a source of data (Micro-blogging) and used TwitterAPI to extract tweets from it.

III. TWITTER APPLICATION PROGRAMMING INTERFACE

This paper deals with performing sentiment analysis on real time tweets and storing tweet scores along with its timestamp. But, the problem is how do we get the data from Twitter. The solution for the problem would be TwitterAPI which provides an interface to collect users' Tweets. A Twitter user's Tweets are also known as status messages. A Tweet can be at most 140 characters in length. Tweets can be published using a wide range of mobile and desktop clients and through the use of TwitterAPI. A user's Tweets can be retrieved using both the REST and the StreamingAPI.[19]

3.1. Rest API

We can access a user's Tweets by using statuses/user timeline from the REST APIs. Using this API, one can retrieve 3,200 of the most recent Tweets published by a user including retweets. [19]

Key Parameters: In each page, we can retrieve 200 Tweets of a user. The parameter max id is used to paginate through the Tweets of a user. To retrieve the next page we use the ID of the oldest Tweet in the list as the value of this parameter in the subsequent request. Then, the API will retrieve only those Tweets whose IDs are below the supplied value.

Rate Limit: An application is allowed 300 requests within a rate limit window and up to 180 requests can be made using the credentials of a user.

Twitter provides the search/tweets API to facilitate searching the Tweets. The search API takes words as queries and multiple queries can be combined as a comma separated list. Tweets from the previous 10 days can be searched using this API. Requests to the API can be made using the method GetSearchResults. Input to the function is a keyword or a list of keywords in the form of an OR query. The function returns an array of Tweet objects.

3.1. Streaming API

Specifically, the statuses/filter API provides a constant stream of public Tweets published by a user. Using the method CreateStreamingConnection .we can create a POST request to the API and fetch the search results as a stream. [19]

Rate Limit: Rate limiting works differently in the Streaming API. In each connection an application is allowed to submit up to 5,000 Twitter userids.

Only public Tweets published by the user can be captured using this API.

Using the Streaming API, we can search for keywords, hashtags and geographic bounding boxes simultaneously. The filter API facilitates this search and provides a continuous stream of Tweets matching the search criteria. POST method

is preferred while creating this request because when using the GET method to retrieve the results, long URLs might be truncated.

We implemented StreamingAPI of TwitterAPI. We implemented two web applications of which one has several number of users and each user is associated with a set of keywords/filters of his interest on which sentiment analysis is performed. Other web application continuously runs which collects the data from twitter using the filters mentioned in users' web application. [19]

IV. NAÏVE BAYES' CLASSIFICATION

Most of the algorithms for sentiment analysis are based on a classifier trained using a collection of annotated text data. Before training, details pre-processed so as to extract the main features. Some classification methods have been proposed: Naive Bayes, Support Vector Machines, K-Nearest Neighbours, etc. However, it is not clear which of these classification strategies is the more appropriate to perform sentiment analysis. We decided to use a classification strategy based on Naive Bayes (NB) because it is a simple and intuitive method whose performance is similar to other approaches. NB combines efficiency (optimal time performance) with reasonable accuracy. The main theoretical drawback of NB methods is that it assumes conditional independence among the linguistic features. If the main features are the tokens extracted from texts, it is evident that they cannot be considered as independent, since words co-occurring in a text are somehow linked by different types of syntactic and semantic dependencies. However, even if NB produces an over simplified model, its classification decisions are surprisingly accurate.

4.1 Strategy

Two different NaïveBayes classifiers have been built, according to two different strategies: Baseline, This is a NaïveBayes classifier that learns from the original training corpus how to classify the three categories found in the corpus: Positive, Negative, and Neutral. So, no modification has been introduced in the training corpus. The second classifier is called Binary which was trained on a simplified training corpus and makes use of a polarity lexicon. The corpus was simplified since only positive and negative tweets were considered. Neutral tweets were not taken into account. As a result, a basic binary (or Boolean) classifier which only identifies both Positive and Negative tweets was trained. In order to detect tweets without polarity (or Neutral), the following basic rule is used: if the tweet contains at least one word that is also found in the polarity lexicon, then the tweet has some degree of polarity. Otherwise, the tweet has no polarity at all and is classified as Neutral. The binary classifier is actually suited to specify the basic polarity between positive and negative, reaching a precision of more than 80% in a corpus with just these two categories.

4.2 Preprocessing

As we will describe in the next section, the main features of the model are lemmas extracted using lemmatization. Given that the language of microblogging requires a special treatment, we propose a pre-processing task to correct and normalize the tweets before lemmatizing them. The main pre-processing tasks we considered are the following:

- Removing urls, references to usernames, and hashtags
- Reduction of replicated characters (e.g.looooveee! love)
- Identifying emoticons and interjections and replacing them with polarity or sentiment expressions (e.g. :-)! good).

4.2 Features

The features considered by the classifier are lemmas, multiword, polarity lexicons, and valence Shifters.

4.2.1 Lemmas (UL)

To characterize the main features underlying the classifier, we make use of unigrams of lemmas instead of tokens to minimize the problems derived from the sparse distribution of words. Moreover, only lemmas belonging to lexical categories are selected as features, namely nouns, verbs, adjectives, and adverbs. So, grammatical words, such as determiners, conjunctions, and prepositions are removed from the model. To configure the feature representation, the frequency of each selected lemma in a tweet is stored.

4.2.2 Multiwords (MW)

There is no agreement on which is the best option for sentiment analysis (unigrams, bigrams ...).The best performance is achieved with bigrams, while the better results are reached with unigrams. An alternative option is to make use of a selected set of n-grams (or multi words) identified by means of regular patterns of PoS tags. Multiword expressions identified by means of PoS tags patterns can be conceived as linguistically motivated terms, since most of them are pairs of words linked by syntactic dependencies. So, in addition to unigrams of lemmas, we also consider multiwords extracted by an algorithm based on patterns of PoS tags. In particular, we
used the following set of patterns:

- NOUN-ADJ
- NOUN-NOUN
- ADJ-NOUN
- NOUN-PRP-NOUN
- VERB-NOUN
- VERB-PRP-NOUN

The instances of bigrams and trigrams extracted with these patterns are added to the unigrams to build the language model. Multiword extraction was performed using our tool which takes care of all these patterns.

4.3 Polarity Lexicon (LEX)

We have built a polarity lexicon with both Positive and Negative entries from a source from Standford University tweet dataset which consists of 10,00,000 tweets with mentioned positive and negative sentiments. On the one hand, it is also used to identify neutral tweets, a tweet is considered as being neutral if it does not contain any lemma appearing in the polarity lexicon. On the other hand, we have built artificial tweets as follows: each entry of the lexicon is converted into an artificial tweet with just one lemma inheriting the polarity (positive or negative) from the lexicon. The frequency of the word in each new tweet is the average frequency of lemmas in the training corpus. These artificial tweets will be taken into account for training the classifiers.

4.4 Valence Shifters (VS)

We take into account negative words that can shift the polarity of specific lemmas in a tweet. In the presented work, we will make use of only those valence shifters that reverse the sentiment of words, namely negations. The strategy to identify the scope of negations relies on the PoS tags of the negative word as well as of those words appearing to its right in the sequence. The algorithm is as follows: Whenever a negative word is found, its PoS tag is considered and, according to its syntactic properties, we search for a polarity word (noun, verb, or adjective) within a window of 2 words after the negation. If a polarity word is found and is syntactically linked to the negative word, then its polarity is reversed. For instance, if the negation word is the adverb "not", the system only reverses the polarity of verbs or adjectives appearing to its right. Nouns are not syntactically linked to this adverb. By contrast, if the negation is the determiner "no" or "none", only the polarity of nouns can be reversed.

V. TIME STAMP BASED SENTIMENT ANALYSIS

Our work on sentiment analysis deals with time i.e. whenever a filter is added by user of web application we use Twitter StreamingAPI to extract the tweets at from that point of time. After each tweet is extracted it is then to Naïve Bayes classifier to get the sentiment score (polarity).The polarity along with the timestamp is stored in database. The procedure is followed with other following tweets. Then a time based analytics is shown to the user based on polarities and timestamps. E.g. :if a user added a filter named 'Narendra Modi'. The web application checks if the filter is already present for the other user, if it is present it starts showing the stats of polarities about 'Narendra Modi' to that particular user. Otherwise, it adds that filter to the Twitter StreamingAPI and then starts extracting tweets about that filter. Then performs Naïve Bayes classification on extracted tweets and displays time based analytics of the classified data.

VI. DESIGN OF OUR WEB APPLICATION

We have designed two web applications for the complete real time sentiment analysis. One of the web application deals with extracting Twitter tweets, performing Naïve Bayes classification on the extracted tweets and inserting sentiment score of tweet and timestamp in database. Other web application deals users and their filters. This web application gets the data of scores and timestamps from database which were inserted into the data base by first web application and displays the time based analytics to their respective users according to their corresponding filters. The web application which deals with extraction, classification of tweets is named as Classification web application and the other which deals with users and their filters is called UI web application.

6.1. UI Application

This web application as the name suggests it deals user interface which at the start up has no users and no filers. When a user registers, he provides his email address and password for his account. Then he can use his these credentials to sign in into his account of UI Application. This Application provides and interface to every user to view all the time based analytics of sentiments of his filters. Initially when user logs in for the first time he doesn't have any filters on his behalf. Application provides settings to add and delete filters by which users can add and delete filters. UI Application also provides means to edit their profile. There are several means of displaying the sentiments of filters like pie chart, bar graph and scaling.

6.2. Classifying Application

This application is very important as it deals with core process of sentiment analysis. At the startup it has no filters. So, it doesn't extract any tweets from twitter. When a user provides UI application with a filter this application starts extracting tweets about the corresponding filter. This it redirects those extracted tweets to Naïve Bayes classifier where it performs preprocessing to remove all the unnecessary information and replaces some information with artificial sentiment words (positive or negative) e.g.: Emoticons with words like '☺' with 'good'. Then the classifier classifies each tweet into positive or negative or neutral. After the complete classification of a tweet, it takes its sentiment score (i.e. 1 for positive -1 for negative and 0 for negative) and timestamp of that tweet and stores it in a database. The same procedure is followed for all following tweets of that corresponding filter and the following filters as well. The database with sentiment score and timestamp will be used by UI application to produce time based sentiment analytics for a filter provided by user.

VII. EXPERIMENTAL RESULTS

In this section, we will present the experimental results that we obtained for a particular user in our web application. User provided us with 2 filters namely AAP (aamaadmi party) and the other python (programming language). There were several tweets about these two filters. We are going to present results when our web application ran for almost 3 hours. There were 567 tweets about AAP and 324 tweets about python when the web application started extracting tweets using Twitter StreamingAPI. There were several tweets that followed. But, for the sake of convenience we freeze these numbers and we will present you with the graphs for those numbers only. Some tweets along with their sentiment scores and time stamps of AAP filter are shown in fig.1.1 and that of python are shown in fig 1.2. The graphs we obtained for these filter are shown in 1.3 and 1.4. we have ran the web application on the next day as well. So there are some changes in scores that are reflected in fig 1.5 and 1.6.

Tweets	scores
AAP nailed it off in delhi	1
AAP bangaya baap	0
AAP jai Ho! ;)	1
this is not a good win AAP	-1
AAP win! BJP out	1
Aaaaaaaaaaaaaaaaaapppp	0

Fig: 1.1

Tweets	scores
Python has never been easy	-1
better programming with python	1
python wins over c#	1
can you do in python?	0
python /c++???	0

Fig: 1.2

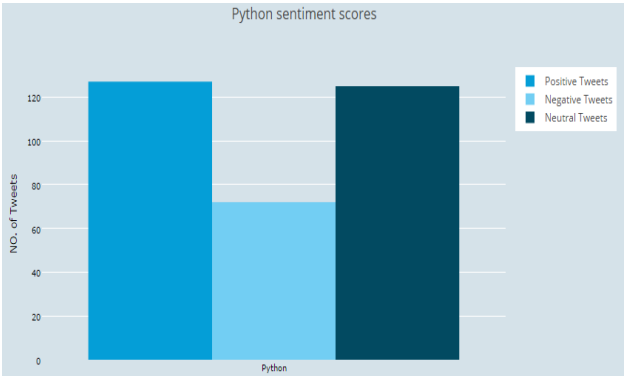

Fig 1.3

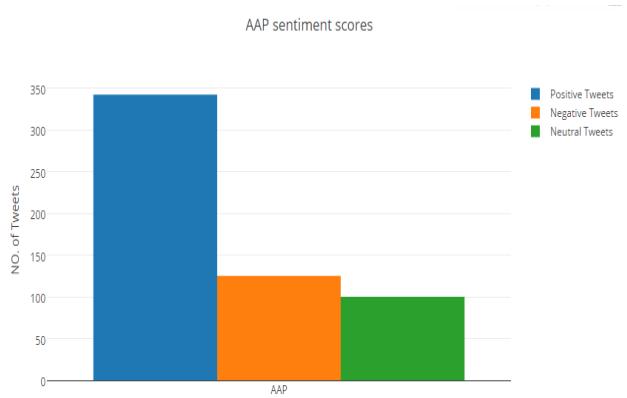

Fig 1.4

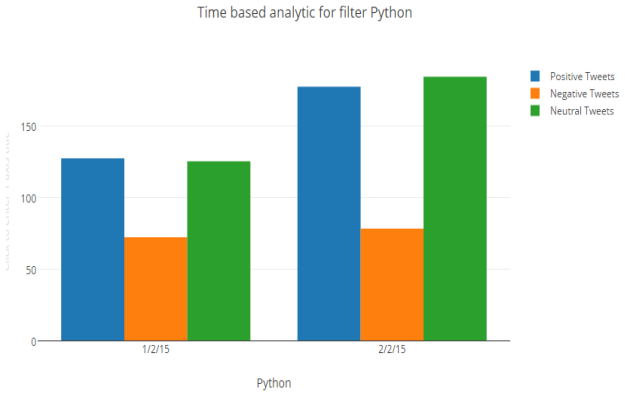

Fig 1.5

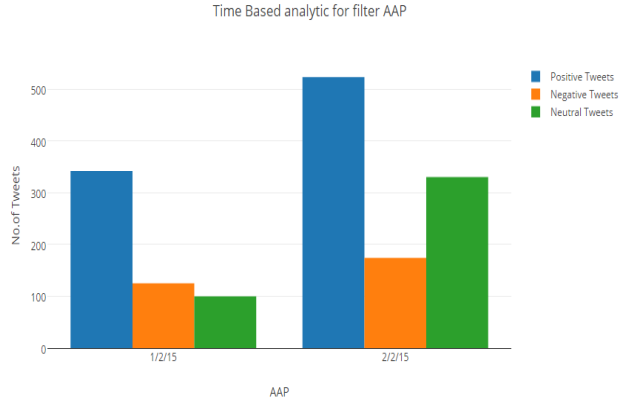

Fig 1.6

VIII. FUTURE WORK

There are several things that can be incorporated in this work which we can do in future one of them would be regional language sentiment analysis. When user has given a filter AAP we came across some tweets which were written in Hindi. We considered them as neutral but in practical we have to perform sentiment analysis using native language. We are expecting good efficiency for our proposed work. Instead of using a basic Naïve Bayes classifier we can use much more efficient and complex algorithm to classify tweets.

IX. CONCLUSION

As we know today's world is becoming a narrower, we get reaction of people for particular products, events, issues very fast on web especially on Facebook and Twitter. Real Time sentiment analysis is very useful to identify and predict current and future trends, product reviews, people opinion for social issues, effect of some specific event on people with time based analytics. ROI and Business Intelligence applications use the sentiment analysis at big organizations like SAP, SAS and TCS. We are concentrating on both objective sentences and subjective sentences so we are going to improve the efficiency and effectiveness of sentiment analysis.

REFERENCES

[1] G.Vinodini and RM.Chandrashekaran, "*Sentiment Analysis and Opinion Mining: A Survey*",International Journal of Advanced Research in Computer Science and Software Engineering, 283-294, Volume 2, Issue 6, June 2012.

[2] Apoorv Agarwal BoyiXie Ilia Vovsha Owen Rambow Rebecca Passonneau, "Sentiment Analysis of Twitter Data",ColumbiaUniversity,New York.

[3] Jalaj S. Modha* Prof & Head Gayatri S. PandiSandip J. Modha, "*Automatic Sentiment Analysis for Unstructured Data*",International Journal of Advanced Research in Computer Science and Software Engineering, 91-97, Volume 3, Issue 12, December 2013.

[4] Pablo Gamallo and Marcos Garcia "*A Naive-Bayes Strategy for Sentiment Analysis on English Tweets*" Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), pages 171–175, Dublin, Ireland, August 23-24 2014.

[5] Harry Zhang "The Optimality of Naive Bayes". FLAIRS2004 conference.(available online: PDF (http:/ / www. cs. unb. ca/ profs/ hzhang/ publications/ FLAIRS04ZhangH. pdf))

[6] Keisuke Mizumoto, HidekazuYanagimoto and Michifumi Yoshioka, "Sentiment Analysis of Stock Market News with Semi-supervised Learning", IEEE Computer Society,IEEE/ACIS 11th International Conference on Computer and Information Science, p.325-328,2012.

[7] Sang-Hyun Cho and Hang-Bong Kang, "Text Sentiment Classification for SNS-based Marketing Using Domain Sentiment Dictionary", IEEE International Conference on Conference on consumer Electronics (ICCE), p.717-718, 2012.

[7] Aurangzeb Khan and BaharumBaharudin, "Sentiment Classification Using Sentence-level Semantic Orientation of Opinion Terms form Blogs", 2011.

[8] Ms.K.Mouthami, Ms.K.Nirmala Devi, Dr.V.MuraliBhaskaran, "Sentiment Analysis and Classification Based on Textual Review".

[9] Online SentiWordNet dictionary source http://sentiwordnet.isti.cnr.it/.

[10] Rudy Prabowo1, Mike Thelwall,"*Sentiment Analysis: A Combined Approach*", School of Computing and Information Technology University of Wolverhampton,Wolverhampton, UK.

[11] Nitin Indurkhya, Fred J. Damerau, "*Handbook of Natural Language Processing",*Second Edition, CRC Press, 2010.

[12] Ronen Feldman, James Sanger,The Text Mining Handbook-Advance Approaches in Analyzing Unstructured Data, Cambridge University Press,2007.

[13] Jiawei Han, MichelineKamber, Data *Mining Concepts and Techniques*, Morgan Kaufmann Publications, 2006.

[14] Ronen Feldman, "Techniques and Application of Sentiment Analysis", Communication of ACM, April 2013, vol. 56.No.4.

[15] Chau, M., & Xu, J. (2007). Mining communities and their relationships in blogs: A study of online hate groups. International Journal of Human – Computer Studies, 65(1), 57–70

[16] Martin, J. (2005). Blogging for dollars. Fortune Small Business, 15(10), 88–92.

[17] Popescu, A. M., Etzioni, O.: Extracting Product Features and Opinions from Reviews, In Proc. Conf. Human Language Technology and Empirical Methods in Natural Language Processing, Vancouver, British Columbia, 2005, 339–346.

[18] Qiang Ye, Ziqiong Zhang, Rob Law, "Sentiment classification of online reviews to travel destinations by supervised machine learning approaches", Expert Systems with Applications 36 (2009) 6527–6535.

[19] Twitter Data Analytics by Shamanth Kumar, Fred Morstatter, Huan Liu August 19, 2013

[20] Bing Liu, *Sentiment Analysis and Opinion Mining*, Morgan and Claypool Publishers, May 2012.p.18-19,27-28,44-45,47,90-101.

[20] Ana C.E.S Lima and Leandro N.de Castro, "*Automatic Sentiment Analysis of Twitter Messages*", IEEE Fourth International Conference on Computational Aspect .of Social Networks (CASoN), p.52-57, 2012.

[21] Gautam Shroff, LipikaDey and Puneet Agrawal, "Social Business Intelligence Using Big Data",CSI Communications, April 2013,p.11-16.

[22] Wikipedia article on supervised machine learning http://en.m.wikipedia.org/wiki/Supevised_learning.

