**IOSR Journal of Engineering (IOSR JEN)** **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
PP 36-38

# Sentiment Analysis Of Tweets Using Semantic Analysis

Snehal Kale [1], Pankaj Kadam[2]
[1](Department of Computer Science &Engg, Thodomal Shahani Engineering College, India)
[2](Department of Computer Science &Engg, Tatyasaheb Kore Institute of Engineering and Technology, India)

***Abstract:*** *In today's world, there is endless stream of data present online and the automated analysis of such data holds a great promise in business analytics for providing a strong support in decision making. This paper looks at the very heart of the concept of sentiment analysis by classifying the tweets with the help of algorithms like Naïve Bayes, Maximum Entropy, and Negation. In this paper, we first preprocess the tweets to remove unnecessary content in tweet; we then extract the adjectives which forms the feature vector, which are also used to find synonyms used in further semantic calculations in aforementioned algorithms. Finally, we calculate Accuracy, Precision, and Recall to compare these algorithms.*

## I. Introduction

In the epoch of social network, people's opinion has become one of the extremely important sources for various services. Opinion on the Internet could be in millions. Sentiment analysis is one of the major focuses in the field of research to extract the immanent data to trail and discern customer opinions.

Considering how crucial such information could be, organizations are often looking for various ways to mine Internet and specially the Twitter where large number of people engage in sharing their opinions and experiences about products and services they have used. However, many a times, we cannot blindly rely on the data extracted from such platforms as they are often expressed in the informal language and can consist of code language, smileys, sarcasm, and so on. And here comes the challenge of making sense of such data, that is, sentiment analysis and unfortunately there is no vast study done in this area. We have seen the successful implementation of Part-of-Speech (POS) tags, feature extraction, negation/intensification, and other features for sentiment analysis, but looking at the nature of our source platform, will it allow us for sentiment analysis on Twitter data? This project will try to find the answer for the same.

No matter how overwhelming amount of users' opinions are available through the web resources, we cannot find the changes required for the system's betterment unless we make sense out of it. In the existing system, we get only reviews and opinions from users that represent a "voice of the users" but are not analyzed properly to improve the online communication.

In the proposed system, we will extract the tweets information from Twitter using web mining techniques and analyze these tweets using sentiment analysis. Sentiment analysis or web mining can be defined as the process of extracting knowledge from sentiments or tweets of users about a topic or problem [1].

We will identify tweets in a large unstructured or structured data then the data collected would be processed to remove useless content. We will then analyze their polarity and will categorize them into different categories basis the Naïve Bayes and Maximum Entropy, and Negation. In this, we are using semantic analysis and classification. Machine learning will be used to analyze the tweets and to tag obtained tweet in a predefined category. The results can be used for various purposes such as guiding decisions to improve the system based on sentiment classification. Sentiment analysis can improve your bottom line.

## II. Material And Methods

In proposed approach, we will use the dataset with tweets and analyze it. We will identify tweets in a large unstructured/structured data then the data collected would be preprocessed. We will then analyze their polarity, and categorization into different categories basis the Naïve Bayes and Maximum Entropy, and Negation will be done. In this, machine learning will be used to analyze the tweets and to tag obtained tweet in a predefined category. The results can be used for various purposes such as guiding decisions to improve the system based on sentiment classification [2] [3].

**Preprocessing of the dataset**

Preprocessing can be defined as a technique that allows transformation of raw/vague data into an understandable format. Data we obtain from different platforms is often unstructured, incomplete, inconsistent, and/or lacking in certain traits and is likely to have ambiguities. Data preprocessing helps resolve these problems. It prepares raw data for further processing. It is highly useful in database-driven applications such as customer relationship management.

In this phase, we clean the data on the following aspects: stop words removal, stemming, punctuation marks, digits and numerals, lowercasing the data, removing repeated words/characters, lastly, after removing these unusable words, the remaining data will be tokenized in tokens.

**Feature extraction**

For sentiment classification, the features are mostly the terms or phrases that influence the sentiment of text. The filtered dataset post preprocessing has a lot of distinctive properties. The feature extraction method extracts the part-of-speech from the dataset. The module contains a part-of-speech tagger for English (identifies adjectives, verbs, and so on.). The broad classification of word classes can be done into open or closed. The maxenttagger class uses a model to perform the tagging task. The words in the sentences are assigned a part-of-speech tags basis their role in the sentence. We will classify the words based on part-of-speech: the adjective (JJ) [4].

**Sentiment and semantic analysis**

In this module, the classification approach followed is similar to objectivity classification, which deals with classifying a tweet or a phrase as either objective or subjective. Once this is done, we check for polarity (only on the tweets that are classified as subjective by the objectivity classification) to find whether the tweet is positive, negative, or both (some researchers include both category and some don't). The following are the machine learning algorithms we will be considering for this classification and get the best result.

**Naïve Bayes:**A Naïve Bayes classifier estimates two things. First, it estimates the probability of each category, independent of any tokens. This is carried out based on the number of training examples presented for each category. Second, for each category, it estimates the probability of seeing each token in that category.
Let's lay out the basic formula to calculate the probability of a category [4].
P(c|x) = (P(x|c)*P(c))/P(x) (1)
Where
P(c|x) denotes the posterior probability of class (category) given predictor (feature).
P(c) denotes the prior probability of class (category).
P(x|c) stands for the likelihood which is the probability of predictor (feature) given class (category).
P(x) denotes the prior probability of predictor (feature).

**Maximum Entropy:**Maximum Entropy maximizes the entropy defined on the conditional probability distribution. Where, c is the class, d is the tweet, and [ λi fi(x,y) ] is a weight vector. The weight vectors decide the significance of a feature in classification [1].

$$p^*(y \mid x) = \frac{\exp\left(\sum_i \lambda_i f_i(x,y)\right)}{\sum_y \exp\left(\sum_i \lambda_i f_i(x,y)\right)} \quad (2)$$

**Negation:**The approach to negation is simple; however, there are many nuances related to negation that need to be considered. The fact is that there are various negators, for example, not, none, nobody, never, and nothing, and other words, such as without or lack (verb and noun), which have an equivalent effect, a few of which may occur at a substantial distance from the lexical term which they affect; a backwards search is required to find these negators, one that is tailored to the particular part-of-speech involved. We assume that for adjectives and adverbs the negation is fairly local, though it is necessary sometimes to look past determiners, and certain verbs [3].

$$F_N(S) = \begin{cases} max\left\{\frac{S+100}{2},\ 10\right\} & if\ S < 0 \\ min\left\{\frac{S-100}{2},\ -10\right\} & if\ S > 0 \end{cases} \quad (3)$$

Where, FN gives the final negation and s denotes the sentiment value from the lexicon using semantic analysis through word net.

## III. Result

In this section, we discuss the findings obtained through Naïve Bayes and Maximum Entropy and compare their relative performances on three parameters: Accuracy, Precision, and Recall [1].

Accuracy is computed as follows:
Accuracy = TP+TN/TP+TN+FN+FP (4)

Precision positive (p) and Precision negative (n) are precision ratios and are computed as follows:
Precision positive (p) = TP/TP+FP (5)
Precision negative (n) = TN/FN+TN (6)

Recall positive (p) and Recall negative (n) are the recall ratios and are computed as follows:
Recall positive (p) = TP/TP+FN (7)
Recall negative (n) = TN/FP+TN (8)

**Table no 1:**Naïve Bayes Classification

| **Performance Measure (%)** | |
|---|---|
| ***Parameters*** | ***Percentage*** |
| Positive Recall | 28.1 |
| Negative Recall | 81.5 |
| Positive Precision | 42.8 |
| Negative Precision | 69.7 |

**Table no 2:**Maximum Entropy Classification

| **Performance Measure (%)** | |
|---|---|
| ***Parameters*** | ***Percentage*** |
| Positive Recall | 95.2 |
| Negative Recall | 9.2 |
| Positive Precision | 22.4 |
| Negative Precision | 87.5 |

**Table no 3:**Accuracy Comparison

| **Methods** | **Accuracy** |
|---|---|
| NaïveBayes | 63.9 |
| Maximum Entropy | 27.8 |

## IV. Conclusion

In this paper, we proposed a set of machine learning techniques with semantic analysis. Based on tweet sentiment score, the words from each sentence and corresponding author will get score by which intention he/she have written that tweets and based on that he/she will get notification. Then we segregated the tweets in different classes, positive, negative, and neutral, based on the sentiments identified in the tweets.

## References

[1]. G. Gautam and D. Yadav, "Sentiment analysis of twitter data using machine learning approaches and semantic analysis," 2014 Seventh International Conference on Contemporary Computing (IC3), Noida, 2014, pp. 437-442. doi: 10.1109/IC3.2014.6897213
[2]. Alessia D'Andrea, Fernando Ferri, PatriziaGrifoni and TizianaGuzzo. Article: Approaches, Tools and Applications for Sentiment Analysis Implementation. International Journal of Computer Applications 125(3):26-33, September 2015. Published by Foundation of Computer Science (FCS), NY, USA.
[3]. Jurek, A., Mulvenna, M. D., & Bi, Y. (2015). Improved lexicon-based sentiment analysis for social media analytics. Security Informatics, 4(9). DOI: 10.1186/s13388-015-0024-x
[4]. P.D. Turney," Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews," Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), Philadelphia, pp. 417-424, July 2002
[5]. Y. Singh, P. K. Bhatia, and O.P. Sangwan, "A Review of Studies on Machine Learning Techniques," International Journal of Computer Science and Security, Volume (1) : Issue (1), pp. 70-84, 2007
[6]. R. Parikh and m. Movassate, "sentiment analysis of user- generated twitter updates using various classi_cation techniques," cs224n final report, 2009
[7]. Go, r. Bhayani, l.huang. "Twitter sentiment classification using distant supervision." Stanford University, technical paper, 2009
[8]. Barbosa, j. Feng. "Robust sentiment detection on twitter from biased and noisy data." Cooling 2010: poster volume,pp. 36-44.