Sentiment analysis using logistic regression algorithm

Yadla Jaswanth[1], Ravilla Muni Sandeep Kumar[1], Ravilla Madhu Sudhan[1],

Mr. Vijaya Kumar S[2] and Mr. Rajagopalam D[3]

[1) Department of Computer Science and Engineering, R.M.K Engineering College, Kavaraipettai, Chennai, India

[2) Associate Professor, Department of Computer Science and Engineering, R.M.K Engineering College, Kavaraipettai, Chennai, India

[3) ETL Solution Architect, Associate Project Manager, DXC Technology India Pvt Ltd, India

Email – yadl16334.cs@rmkec.ac.in

Abstract. ***Sentiment Analysis is a sub-field of Natural Language Processing. This is characterised as a method of recognizing and classifying opinions from a text document and is helpful in determining user's intention for specific subject is neutral, negative or positive. It is also termed as Opinion Mining. The motive of our proposed work is to detect hate speech in tweets. In our case, we consider a twitter post to be a hateful speech if it has a negative meaning. Hence, our objective is just to categorize negative tweets from overall tweets.***

1. Introduction

Understanding how customers respond to a product is very critical for further development or growth of a business. If the customer gets satisfied with the service offered by the company, then it is regarded as a company's success. Analysing customers means finding their opinion on a specific topic, this is the point where sentiment analysis plays a major role. Sentiment analysis is not only useful in the business domain, people often give their opinions on movies which is very important for filmmakers to identify the audience response [1]. In recent times, usage of social media and social networking sites are increasing exponentially. We have taken Twitter social media platform as a main source for viewing sentiments. The reason for choosing twitter is that, most of the tweets expressed are opinionated. A number of research works are underway in examining the emotions to classify individual behaviours, responses or opinions [2].

There are 3 different ways of mining a sentiment, they are described this way [3]:

(1) Sentence level sentiment analysis: Every statement is categorized as neutral, negative or positive at this point
(2) Aspect level sentiment analysis: Documents are classified as neutral, negative or positive based on certain aspects.
(3) Document level sentiment analysis: The whole document is categorized as neutral, negative or positive.

Three different kinds of techniques for classifying sentiments are proposed in [10]:

(1) Rule based method: At this stage, Lexicons and predefined principles are used to identify the sentiment of the users on a particular topic.
(2) Machine learning based method: In this approach we build a machine learning classifier which gives polarity of subject based on the trained data.
(3) Hybrid method: This approach is a combination of both the rule based as well as machine learning based approach.

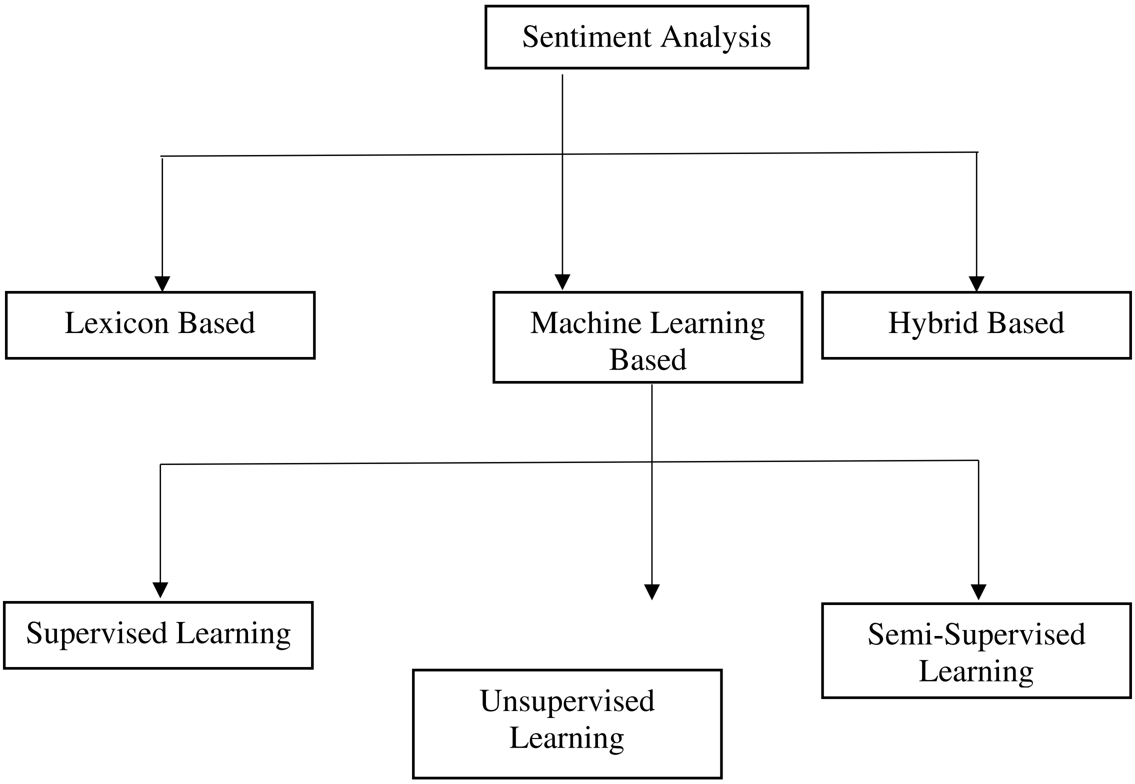

Figure 1. Depicts various sentiment classification methods [10].

2. Literature Review

WordNet [4] is utilised for identifying sentiment associated with a term in various ways [3]. Distance metric was generated on WordNet and the sematic orientation of adjectives was found by them.

In 1970, Ekman et al. [5] did immense research in multiple expressions on face and expressed that these are adequate for identifying emotions.

Akba et al. [6] used collection of functionality-focused on information gain and chi-Square metrics upon completion of the lemmatization and stemming process, the insightful characteristics are selected. The tests performed have shown that the correlation of feature engineering measurements with support vector machine classifiers has increased relative to previous research.

Twitter corpus was developed by gathering twitter posts from application programming interface provided by twitter and interpreting them by making use of emojis [7]. Sentiment analysis model was constructed by considering this corpus.

A technique to get specifications such as battery, processor, camera for a specific product was developed in [8]. The main technical aspects of a product are found and classified. Depending on whether neutral, negative or positive scores were assigned for each and every specification. By combining all the scores of independent features, the overall rating of a product was identified.

For categorizing the feedback, an upgraded method from Support Vector Machine was proposed in [9]. Depending on the words associated with emotions, SentiWordNet assigned the sentiment scores. By changing these score they developed a modified model.

3. Proposed System

Our proposed system is to identify the hate speech in the extracted tweets. We will categorize the tweets as hate speech if a negative sentiment is associated with a tweet. So, our task is to classify negative tweets from the overall tweets. Initially, we classify whole data set into a two types (Training and Testing sets). The former one contains 3 columns namely id, label and tweets. Label column contains binary values such as 0 and 1. We proceed by allocating labels to our training set where label 1 denotes a negative tweet whereas label 0 denotes not a negative tweet. Now, we build a model using a logistic regression algorithm. After training this model, it predicts the negative and non-negative tweets in the testing dataset by labeling them with 0 or 1. In our case, we used the f1 score as an evaluation criteria.

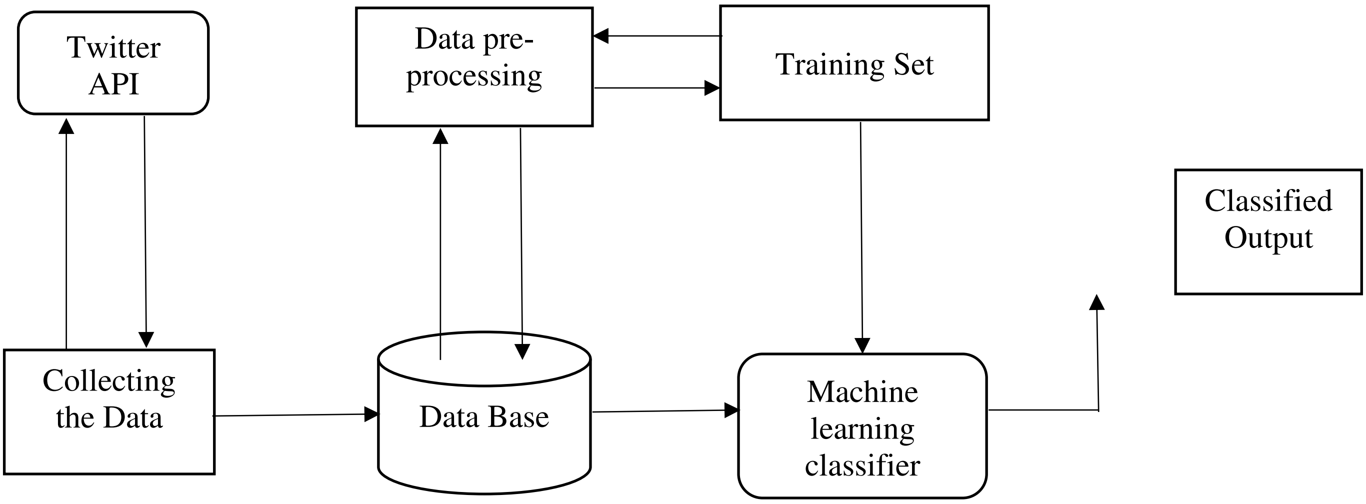

Figure 2. Explains how the data flows between each phase.

4. Methodology

Sentiment classification involves following stages:

4.1. Text pre-processing and cleaning

This is considered as important phase because it prepares the unstructured data for processing. If we have not performed this step properly, then there is a high probability that we may deal with scattered and inaccurate information. So, the goal of performing this phase is to remove unrelated text from the twitter posts like special characters and punctuation which does not add any value to calculate sentiment

(i) Removing Twitter Handles: When collecting the tweets all the twitter accounts are concealed because of privacy issues. So, we wipe all the twitter account usernames(@user) out of all the gathered tweets since these are not necessary for finding the sentiment.

(ii) Removing punctuations, numbers and special characters: In this step, excluding hashtags and characters we replace every other thing with spaces.

(iii) Removing Short words: In this step, we should take certain measures in choosing the word length meticulously to remove. So, we choose the word length having length of 3 or less to remove from the tweet. For example, words like “hmm”, “oh” which do not have any sentiment.

(iv) Tokenization: Tokens are defined as individual words and the method of breaking a string into tokens is called tokenization. For example, let us take a tweet “He played good cricketing shots in that match”. After Tokenization it looks like

[‘He’, ‘played’, ‘good’, ‘cricketing’, ‘shot’, ‘in’, ‘that’, ‘match’]

(v) Stemming: The process of removing suffixes from a word is performed in the stemming phase. For example, sing, singer, singing, sings all come under the root word sing.

4.2. Visualisation from tweets

For this purpose, we use word cloud as a means for representing the key words from the above classified tweets. In this technique, maximum number of used words are displayed in bigger size and least occurring one in the tiny size.

Figure 3. Word cloud representation.

4.3. Feature extraction technique

In order to examine pre-processed information, these must be transformed into further stage. There are many ways available for extracting features. They are

(1) Bag of words Approach: This technique is used to show the content into mathematical format. Let us assume a corpus K has R documents {r1,r2,r3...rN} and M distinctive tokens taken from the corpus K. The M tokens generates a matrix in R × M format. The rows in matrix hold the occurrence of tokens in R(i).

Example: Consider there are 2 documents:
R1: Kiran is a good student. He is a hard-working student.
R2: Kim is a smart student.
The list that was generated composes all the unique tokens in the corpus K.
['Kiran', 'good', 'student', 'work', 'hard', 'Kim', 'smart']
Here, R=2, M=7.

Table 1. Matrix Representation of above example.

	Kiran	good	student	work	hard	Kim	smart
R1	1	1	2	1	1	0	0
R2	0	0	1	0	0	1	1

The classification model is built by making use of information from above table.

(2) TF IDF Method: TF is denoted by term frequency whereas IDF is denoted by inverse document frequency. This method is also dependent on frequency but can be distinguished from the former method, in such a way that it sees for occurrence of a word in the whole corpus. This approach assigns lowest weight to the most repeated words in the corpus and give preference to words which have occurred very less times in corpus.

The significant phrases used in TF-IDF are:

- TF = (the count of the term t appears in a text document)/(Number of terms in the document)
- IDF = log(D/d), where, D is the number of documents and the number of documents a term t has appeared is referred to as d.
- TF IDF = TF*IDF

5. Algorithm

In our project, we make use of the Logistic Regression algorithm to build the model. It identifies the probability of occurrence of an event by fitting data to a logit function.
The equation used in the algorithm is:

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta(\text{num}) \quad (1)$$

Here, If the log(p/(1-p)) is greater than zero, then the success ratio is every time appears to be greater than half of the 100 percent.

$$\text{F1-Score} = 2 * \left(\frac{A*B}{A+B}\right) \quad (2)$$

F1-Score can be a defined as harmonic mean of A and B. The F1-Score of 0.54 is observed for validation set by using the Bag-of-words approach. Now, we will use the TF-IDF approach for calculating F1-Score for the same model and it appears to be 0.558 for validation set. Overall, by considering TF IDF technique, we can observe increase in the validation score. Here, A and B are represented as precision and recall respectively.

References

1. Svetlana Kiritchenko, Xiaodan Zhu, Saif M Mohammad Sentiment Analysis of Short Informal Tests. Journal of Artificial Intelligence Research 50 pp. 723-762 2014
2. Walaa Medhat a, * Ahmed Hassan b, HodaKorashy b Sentiment analysis algorithm and applications: A survey in Ain Shams Engineering Journal Page no – 1094
3. Kamps J, Marx M, Mokken R J and De Rijke M "Using wordnet to measure sematic orientations of adjectives"
4. Fellbaum C "Wordnet: An Electronic lexical database (language, speech, and communication)"
5. Ekman P "Universal facial expressions of emotion," Culture and Personality: Contemporary Reading (Chicago), pp. 151-158 1974
6. Akba F, Ucan A, Sezer A and Server H, "Assessment of feature selection metrics for sentiment analysis: Turkish movie reviews", in 8th European Conference on Data Mining, Vol. 191, pp. 180-184 2014
7. Pak A and Paroubek P "Twitter as a corpus for sentiment analysis and opinion mining", in Proceedings of LREC. Vol 10
8. Vamsee Krishna Kiran M, Vinodhini R E, Archanaa R and Vimal Kumar "User specific product recommendation and rating system by performing sentiment analysis on product reviews", 4th International Conference on Advanced Computing and Communication Systems (ICACCS), Vol 207 pp.1-5
9. J. Bhaskar, sruthi K and P. Nedungadi, "Enhanced sentiment analysis of informal textual communication in social media by considering objective words and intensifiers", International Conference on Recent Advances and Innovations in Engineering (ICRAIE), Jaipur, pp. 1-6 2014
10. Madhoushi Z, Hamdan A R and Zainudin S "sentiment analysis techniques in recent works", London: Science and Information Conference (SAI), pp.228-291 2015