International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-6, April 2020

Prediction of Social Trends using Twitter

S. Babu, Kushagra Singh Bisen, Himanshu Chaubey

Abstract*: *In this study project, we deal with the application of sentimental analysis on a dataset consisting of tweets. We implement a number of machine learning and deep learning frameworks to perform the analysis. In the end, we use a majority based method to show 4 of our best models for the dataset to achieve the best result.

Keywords: Tweets, Sentimental Analysis, Data Preprocessing, Machine Learning, Deep Learning, Feature Extraction.

I. INTRODUCTION

Twitter is a widely used social networking platform where users create accounts and exchange information in the form of words, website links, images and videos. This information is known as "tweet". A "tweet" can serve as a medium for the users to showcase their thoughts and feelings about topics in the country, personal hobbies and to keep up with their friends and family. Consumers, as well as product producers, use sentiment analysis to predict the market and to gather insights about a specific topic or their product they've launched in the market. In this project, we attempt to perform sentiment analysis over "tweets" by implementing various machine learning and deep learning algorithms. We try to classify the polarity of the tweet, whether it is positive or negative. In certain cases, if the tweet has both the characteristic of positive and negative; we use the prominent characteristic as the deciding feature for classification. In the project, we use a dataset containing lakhs of tweets, then differentiate them into testing and training dataset. The preprocessing part comes where we convert emoticons, hashtags and usernames into a standard form useful for the analysis. We extract useful features from the data like unigrams to provide the proper 'representation' of the "tweet". After the data processing part, we further implement various machine learning and deep learning. We don't rely on a single framework, rather use more than 4 and then report the best framework for the specific dataset as there is a chance that a single model can't provide high accuracy. We report the final findings at the end. Sentimental Analysis in a method has a lot of applications like in sectors such as Online Commerce to personify the experience for the user and show the best recommendations for him.

Revised Manuscript Received on April 30, 2020.
***** Correspondence Author
Dr. S. Babu*, Department of Computer Science and Engineering, SRM-IST,Kattankulathur,Kancheepuram,India.
Kushagra Singh Bisen, Department of Computer Science and Engineering,SRM-IST,Kattankulathur,India.
HimanshuChaubey, Department of Computer Science and Engineering,SRM-IST,Kattankulathur,India.

It is also used to check the Voice of Market or VOM, which deals with finding how the people react to the advancement of the new product being introduced to them. Brand Reputation Management is also an interdisciplinary sector which deals with addressing the concern of what people think of the brand and what they like and what they do not. Sentiment analysis is also used by the government to check the reaction of people and their reaction towards a particular candidate.

II. LITERATURE SURVEY

1. Earlier Approaches

2009 - Go, Bhayani and Yuang

They used to get a query and then classify the tweet into positive and negative. They used a method to classify and get tweets directly from twitter. To clarify the mood of the tweet, they used implemented methods to classify emotions of the tweet into positive or negative by using emoticons. Smiling emoticons like :), :-), :D, were labelled as positive and emoticons like :(, :'(, :-(were labelled as negative.
They try various features such as bigrams, unigrams, and train them on various machine learning models such as Naive Bayes, Support Vector Machines and then compare it to another classifier by counting the number of positive and negative words from it. In the end, they report that bigrams aren't the deciding factor and support vector machines show the best results.

2010 - Pak and Paroubek

Their problem statement was the fact that informal and creative language people use and the slangs which vary location to location make the process of sentimental analysis very tough. They use previous work done in the sector to clarify and build their own classifier. They used Edinburgh twitter corpus to check the most used hashtags on twitter. They at first, manually classify these tweets and then classify them into tweets. Apart from using n-grams and Part-of-Speech features, they built a feature set from the previously existing dictionary. They report that the best features are visible with an n-gram based features.

2012 - Saif, He and Alani

They identify the tweet to be a person, an organization etc. They also show that the removal of stop words isn't a required step and can have a not required effect on the classifier. All the processes involve an n-gram feature. It is unclear if Part-of-Speech is useful or not. To improve accuracy, some of the features involve a different method of feature selection by using knowledge about micro-blogging. They use concepts such as sentimental analysis, like stemming, two-step classification and negation detection and dealing with the scope of negation.

Negation detection is a process which is often discussed in the sentimental analysis. Negation words like “not”, “never”, “Nah” and “no” can drastically change the meaning of the tweet and the related sentiment value attached to them. If these words are present, the meaning of the nearby words will change a lot. Such words are known as the scope of negation. The scope of negation can take a cue from the nearest available punctuation. They find negation cues from left to right for the tweet and also find a distance to the nearest cue in left as well as right.

2011 - Wilson and Moore

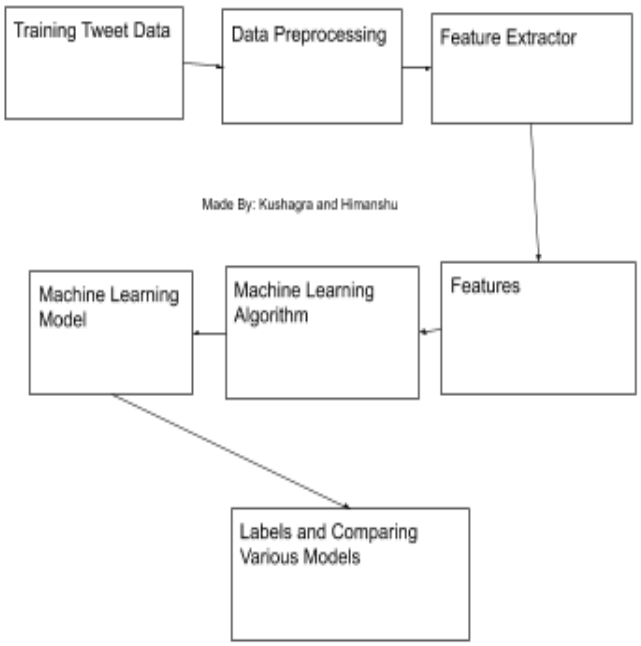
They deal with the linguistic features for detecting the sentiment of Twitter messages. They check if the existing lexical methods are useful and as well as checking features such as microblogging. They undertook a very supervised approach to the problem but also leverage the existing features such as the existing hashtags in Twitter for building Twitter Data.

Inference from the survey

The paper by Bhayani in 2009 tries to showcase various features such as unigrams, bigrams and Part-of-Speech to train the classifier on a lot of machine learning algorithms, such as Naive Bayes and Maximum entropy and vector machines. Bigrams alone are not enough to classify and Naive Bayes gives the best result. The 2010 paper shows that normal slangs and the language we use as locals with colleagues disrupts the sentimental value of the tweet. They use previously existing work in the sentiment analysis than building their own classifier. We can see that there has been enough work done in the field but still, there is a lot of work and research to be done.

III. PROPOSED WORK

BLOCK DIAGRAM:



The methodology we are implementing in our project is divided into various separate modules and processes. At first, we will make a dataset of tweets which can either be made by self or downloaded from Kaggle, just keep it in mind that the dataset should be original and should be truly labelled showing the tweet id, sentiment value and the tweet itself. Once we get the dataset prepared, we will separate it into 3 parts, the training, testing and the validation dataset. We further perform preprocessing in the data by dealing with individual features and then implement machine learning and deep learning frameworks to check and see the accuracy of those frameworks and thus predict the best classifier from all of it. Feature extraction is a very important feature in the implementation of the project. From the perspective of Sentimental Analysis, there are a lot of features which matter, such as the length of the tweet, the language of the tweet in which it was written, the magnitude of data available using the twitter API to create a generic classifier.

IV. IMPLEMENTATION

Data Preprocessing-

Tweets that people do regularly are of raw nature. This is mainly because of the raw nature of people and the way they express their feelings. Tweets have a special way of expressing feelings which are through emoticons, retweets and user mentions. Thus, the twitter dataset has to be classified and processed such that it can be used and get input into the machine learning models. We do some general preprocessing such as:

- Convert the existent tweet into lower case.
- Replace two or more dots(.) with space.
- If the tweet contains spaces, strip it at the beginning and the end.
- Replace two or more space with a single space.

URL preprocessing:

Users often share more than one type of links in their tweets, thus we need to clarify and separate the URL by assigning it a URL based identifier.

User Handles:

Each user in twitter has a specific username available for him. Now to use it for analysis, we omit the username and separate and replace it by USER_NAME.

Emoticons:

Emoticons are very important as they decide what’s the sentiment of a specific tweet and then classify it to positive or negative or what’s more prominent. Thus, we self decide the emoticons and then designate EMO_POS and EMO_NEG to it.

Hashtags:

Hashtags are unspaced phrases, which are used to track or conribute to a a discussion. We, in the implementation, remove the hashtag and store it like a string.

Retweets:

Retweets are the tweets already sent by some user and are then shared by another, they start with the letter RT and are of no use in classification, so we will drop the “RT” from the beginning.

Feature Extraction:

Now, features are very important when it comes to the implementation part,

in this case we use two features, namely Unigram and Bigram. Unigrams are he simplest and the most commonly used tool for feature extraction in the form of simple words or tokens. There are many such unigrams in the dataset but we only use top 15000 because most of it can be considered as noise. The plot between log(Frequency) and log(Rank) is seen with the trendline being log(Frequency) = -0.78log(Rank) + 13.31 .

Bigrams are word pairs which occur in succession. They are of good use in classification as they show the meaning, *The dessert was no good.* This dictates the real meaning around the tweet.

Feature Representation:

After we get both unigram and bigram,it is our job to convert them into sparse and dense vector matrix for their specific classification methods.

Depending on the fact that if we are using both unigrams and bigrams in our dataset, there are supposed to be 15000 bigrams and 25000 both unigrams and bigrams. The positive and negative value of it depends upon the frequency and presence. Handling memory loss issues are an important application of feature representation.

Dense vector representation deals with 90000 features who are indexed and assigned a rank.

Classifiers:

Naive Bayes is a method through which we assign and classify texts, in this scenario c' is the class assigned to the tweet t, such that

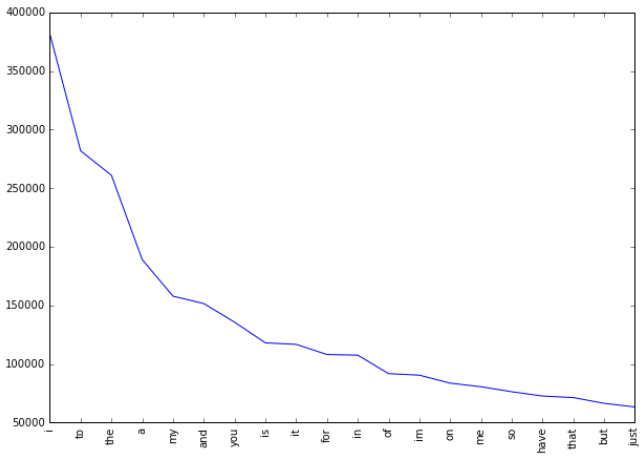
$$c' = \text{argmax } P(c|t)$$

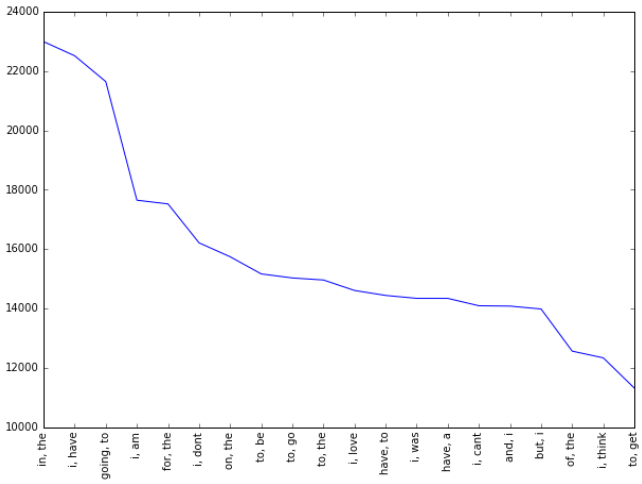
which can be used through the likelihood estimates.

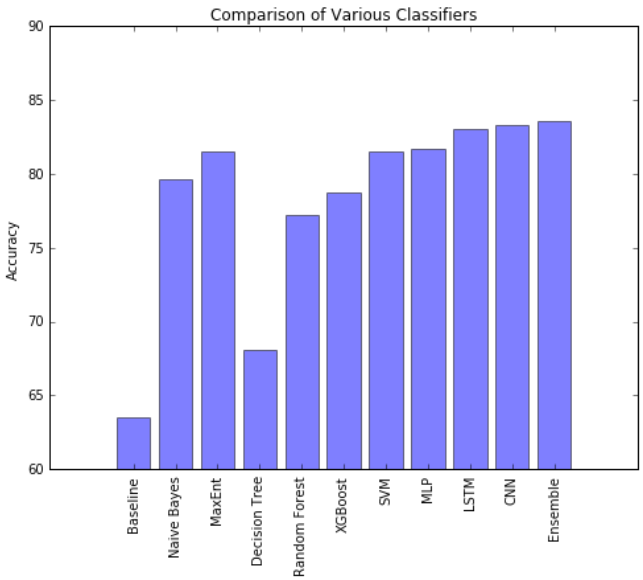
Decision Trees are a classifier in which each node of the tree represents the final dataset. Leaf nodes show the final classes of the data. It is a supervised classifier which uses labels to form a decision tree.

Random forest is an ensemble learning algorithm which can be both used for classification and regression. It generates a multitude of those trees depending upon the addition of the decision trees. Support Vector Machines or SVM are non-probabilistic binary linear classifier. For a dataset in between the points (x,y) we would like to find a maximum margin hyperplane. Convolutional neural networks are a type of neural networks involving various layers such as convolutional layers who can interpret the spatial form of data. Kernel here, in this case, is a 2D window, Now for the implementation part, we split our dataset and then leave 10% of the dataset just for the validation set. Now, we use sparse vector representation in the case of Naive Bayes, Random Forest, Decision Trees, Support Vector Machines and Dense Vector Representation in the case of Convolutional Neural Networks. In order to improve accuracy even further, we make a simple ensemble model. We extract 500 best features from our nest performing classifiers for each tweet which is represented by a 600-dimensional feature vector. We classify votes using SVM and then take the majority vote with 5 models.

V. RESULTS







VI. CONCLUSION

The tweets we were provided are a mixture of emoticons, URLs and user handle mentions. Before training, we make them suitable for feeding into our models by preprocessing. We implemented several machine learning algorithms such as Naive Bayes,

Decision Tree, Random Forest, Support Vector Machine and Convolutional Neural Networks to classify the orientation of tweet if it is positive or negative. We make two types of features namely unigrams and bigrams for classification and see that feature vector with bigrams optimises the accuracy. The extracted feature is represented in either sparse or dense vector. Presence in the sparse vector is better than efficiency. We finally make the best of our models and prediction with the most accuracy.

FUTURE WORK

- Handling Emotion Ranges by classifying the tweets into more than just positive and negative like classifying them into -2 to +2. “This is good” is positive but “This is extraordinary” is more positive than it.
- In preprocessing, we discard features like commas and question mark which maybe helpful and we can optimize our model for such purpose.
- We can make a analyzer for other languages such as Hindi.
- We can further investigate if support vector machines improve accuracy.
- Implementing Semantic Analysis to improve the accuracy.

REFERENCES

1. Pak, Alexander, And Patrick Paroubek. "Twitter As A Corpus For Sentiment Analysis And Opinion Mining." Lrec. Vol. 10. 2010.
2. Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision. Processing, pages 1-6, 2009.
3. Alexander Pak and Patrick Paroubek. Twitter as a corpus for sentiment
4. John Ross Quinlan. C4. 5: programs for machine learning, volume 1. Morgan kaufmann, 1993.
5. Steven Bird, Ewan Klein, and Edward Loper. Natural language processing with Python. " O'Reilly Media, Inc.", 2009.