KATHMANDU UNIVERSITY

SCHOOL OF ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Mini-Project Report on

“Sentiment Analysis”

[Code No. COMP 473]

(For the partial fulfillment of this course in 4th year 2nd semester in Computer Engineering)

Submitted by:

Yachana Aryal (03)

Submitted to:

Dr. Bal Krishna Bal

Head of Department

Department of Computer Science and Engineering

Submission Date: 31/07/2020

Table of Contents

Chapter 1: Introduction:

1.1 Problem Definition:

Sentiment analysis is the process of analyzing people's feelings or emotions towards certain things. It can be used in businesses to identify customer sentiment toward products, brands or services in online conversations and feedback. Here, we use tweets to find out the sentiments of people expressed in that particular tweet. The tweet is given as an input to the system and in response the system gives out the emotion represented by the tweets. We divide tweets into positive and negative tweets.

Sentiment Analysis is a NLP based system which categorizes the emotions of the author on a topic that is being written about into variety of sentiments. It consists of a training data set which trains the model using a classification algorithm. In classification algorithm, each dataset is associated to a sentiment for both training and testing. It defines two lists of polarized words (e.g. negative words such as bad, worst, ugly, etc. and positive words such as good, best, beautiful, etc.) and counts the frequency of each word. If the frequency of positive word is greater than that of negative word, the system returns a positive sentiment otherwise it returns a negative sentiment.

1.2 Objective:

The objectives of the sentiment analysis are:

- To identify the sentiment in short texts like tweets.
- To determine what types of words are used mostly in the tweets.
- To understand the attitude, opinion and emotion about certain things.

1.3 Motivation:

With the popularity of twitter, people are constantly using it to express their feelings or thoughts about certain things. So with the extraction of the type of emotion every tweet represents, we could use it in identifying the current trend or in reviewing opinions about services or products of companies. It could also be used to determine or predict the winner of the election to be held on the basis of people's opinion about the leaders. Furthermore, it could be used to know the mental state of people whether they are constantly tweeting positive tweets or negative. It has a vast range of applications.

Chapter 2: Related Works:

2.1 Case I:

Detecting political activism in the twitterverse:

This project analyses the social media conversations in search of the public opinion about an unfolding political event that is being discussed in real-time for example: presidential debates, major speeches, etc. It distinguishes between two groups of participants: political activists and the general public. It uses a supervised machine-learning approach with the acquired labeled data from mono-thematic Twitter accounts to learn a binary classifier for the labels “political activist” and “general public”. While the classifier has a 92 % accuracy on individual tweets, when applied to the last 200 tweets from accounts of a set of 1000 Twitter users, it classifies accounts with a 97 % accuracy. This work demonstrates that machine learning algorithms can play a critical role in improving the quality of social media analytics and understanding, whose importance is increasing as social media adoption becomes widespread.

2.2 Case II:

Investor Classifier:

This involves sentiment analysis on tweets of the potential investors. Investors are assumed to be sentiment driven. Sentiment analysis is done on the tweets pulled from some selected investors’ twitter feeds. They assign positive, negative and neutral sentiment scores to the ticker symbols from the pulled tweets by identifying “bad”, “not good”, “great” words in the tweets. For processing, the unstructured text in the tweets “Microsoft Azure” analyser tool is used. Through sentiment analysis, a particular user can understand the social sentiment score of a ticker symbol based on the discussions of key investors and make an informed decision about which stock to invest.

Chapter 3: Dataset

We used the dataset of sample tweets from the NLTK package for NLP. We used 5000 tweets labeled as positive tweets, 5000 negative tweets labeled as negative tweets and 2000 tweets as non-labeled tweets. The positive, negative and non-labeled tweets are merged into the dataset and shuffled randomly. Then, we divided the dataset into training and testing dataset in the ratio 70:30 where 70% of dataset are used for training and 30% for testing.

Chapter 4: Methods and Algorithms Used

4.1 Proposed System Design:

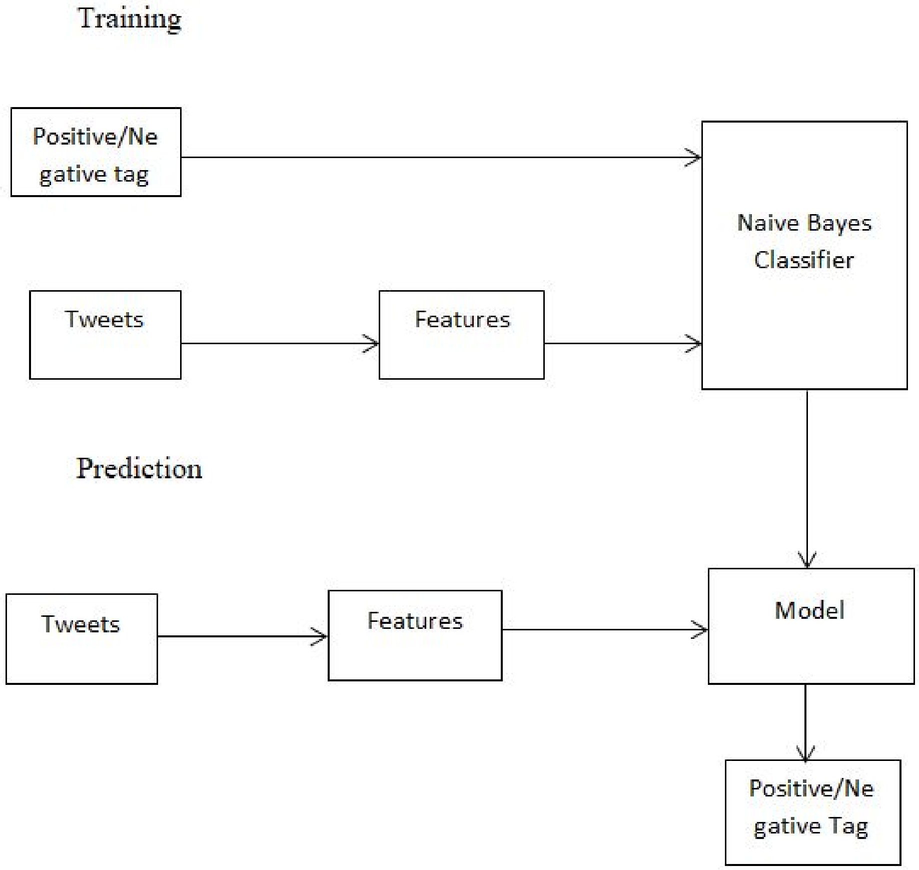


Fig 1: System design of Sentiment analysis

This is the proposed design of the system. The system consists of training part and prediction part. In training part, the system is provided with the labeled training dataset which consists of tweets with corresponding tags. The model is then built using Naive Bayes classifier which is a supervised classifier. After training the model, it is tested using test dataset and used for prediction on the unobserved data.

4.2 Data Tokenization and Normalization:

Firstly, the whole text is tokenized into smaller parts called tokens using a method from NLTK. Then, the data is normalized which includes stemming and lemmatization. For this, the tags for each token were generated in the text using the function pos_tag, and then each word was lemmatized using the tag with the help of wordnet database which helps to determine the base word.

4.3 Removing Noise and Visualizing data:

After the data is tokenized and normalized, the noises like hyperlink, punctuation and special characters are removed using regular expressions. Also, stop words like the, and etc. were removed from the text using stopwords method from nltk library and converted into lowercase using removenoise function. Then for the data visualization, the word density is determined using FreqDist class of NLTK which shows the most common words in a particular dataset.

4.4 Preparing Data for the model:

Before feeding the data to the model, the tweets from a list of cleaned tokens are converted to dictionaries with keys as the tokens and True as values. And the dataset is split into training and testing data in the ratio of 70:30. Now, the training data is ready to be fed to the model. The model we are going to use is Naïve Bayes Classifier.

4.5 NLTK Library:

The Natural Language Toolkit, or more commonly NLTK, is a suite of libraries and programs for symbolic and statistical natural language processing (NLP) for English written in the Python programming language.

4.6 Naive Bayes Classifier:

A Naive Bayes classifier is a probabilistic machine learning model that's used for classification task which is based on Bayes' Theorem**.**

Bayes' Theorem finds the probability of an event occurring given the probability of another event that has already occurred. Bayes' theorem is stated mathematically as the following equation:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Chapter 5: Flowchart

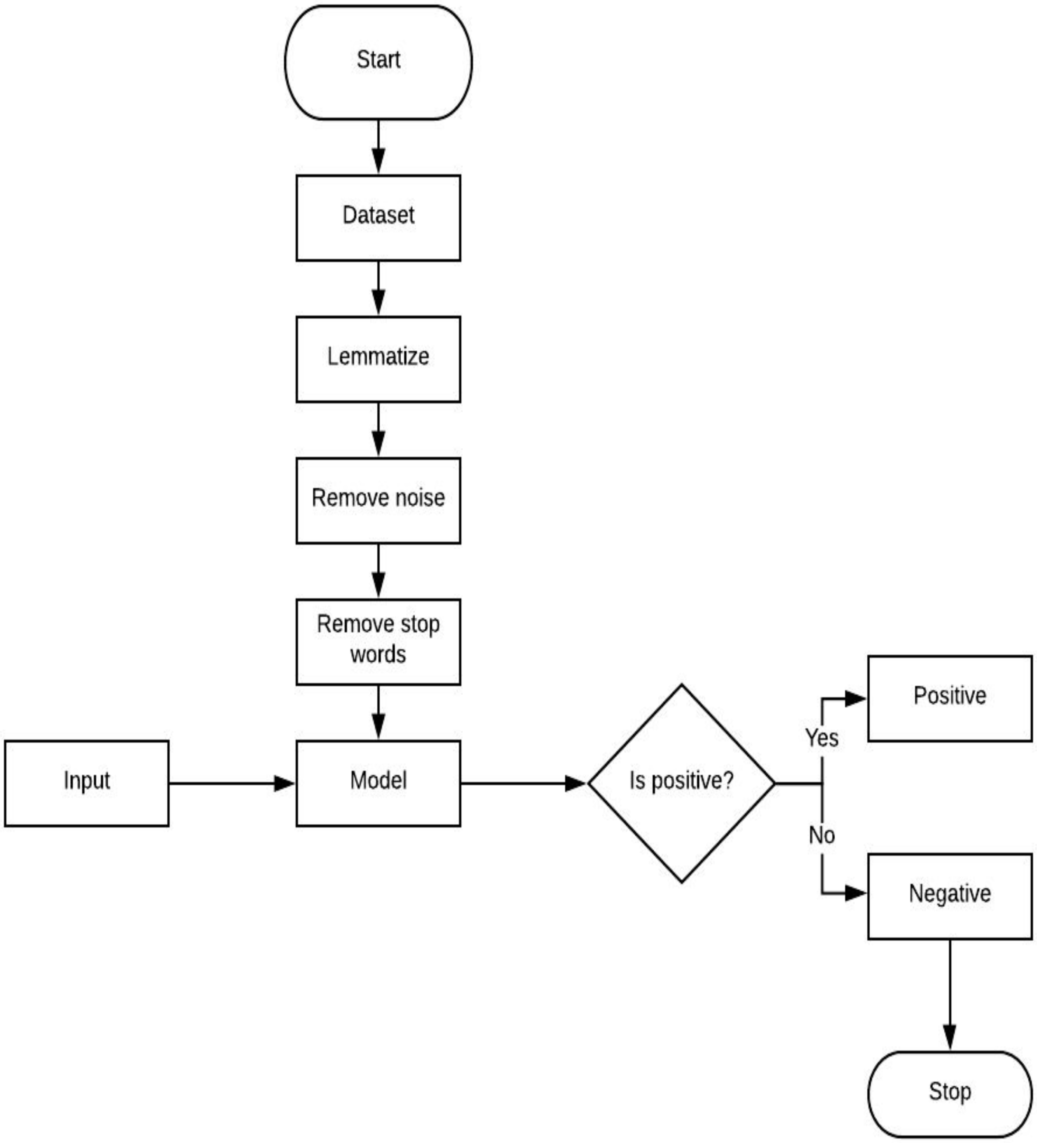
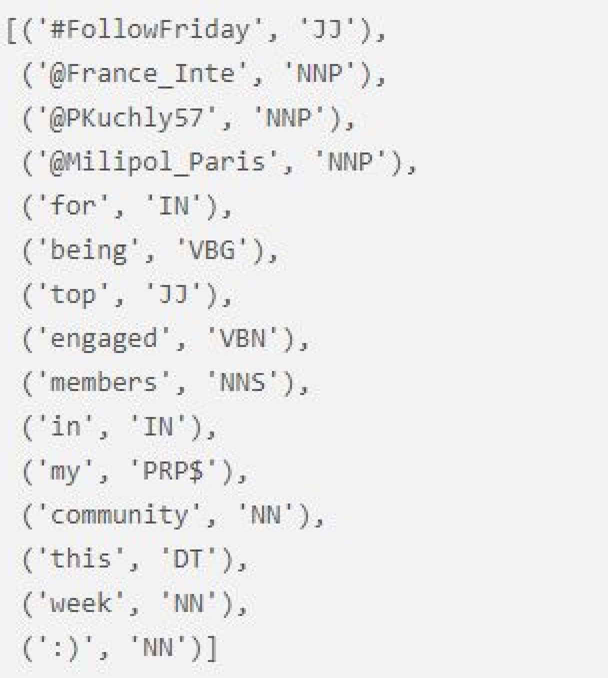


Fig 2: Flowchart of Sentiment Analysis

Chapter 6: Experiments

The experiments done in the projects are as follows:

i. POS tagging : Tagged each word in the dataset with the corresponding parts of speech.

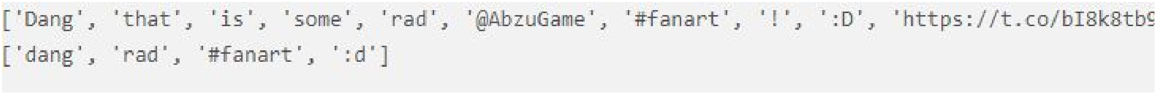


```
[('#FollowFriday', 'JJ'),
 ('@France_Inte', 'NNP'),
 ('@PKuchly57', 'NNP'),
 ('@Milipol_Paris', 'NNP'),
 ('for', 'IN'),
 ('being', 'VBG'),
 ('top', 'JJ'),
 ('engaged', 'VBN'),
 ('members', 'NNS'),
 ('in', 'IN'),
 ('my', 'PRP$'),
 ('community', 'NN'),
 ('this', 'DT'),
 ('week', 'NN'),
 (':)', 'NN')]
```

ii. Lemmatization : On the basis of parts of speech tagged to the words, each word is changed into its base form.

iii. Removing noise from the data: The hyperlinks twitter handles in replies, punctuation and special characters are removed using regular expressions.

iv. Remove stop words using inbuilt function from nltk corpus and converted into lowercase



```
['Dang', 'that', 'is', 'some', 'rad', '@AbzuGame', '#fanart', '!', ':D', 'https://t.co/bI8k8tb9
['dang', 'rad', '#fanart', ':d']
```

vi. Converting tokens into dictionary to feed into the model

vii. Splitting the dataset into training and testing data in the ratio of 70:30

viii. Using Naïve Bayes classifier to classify the sentiments.

Chapter 7: Evaluation of results

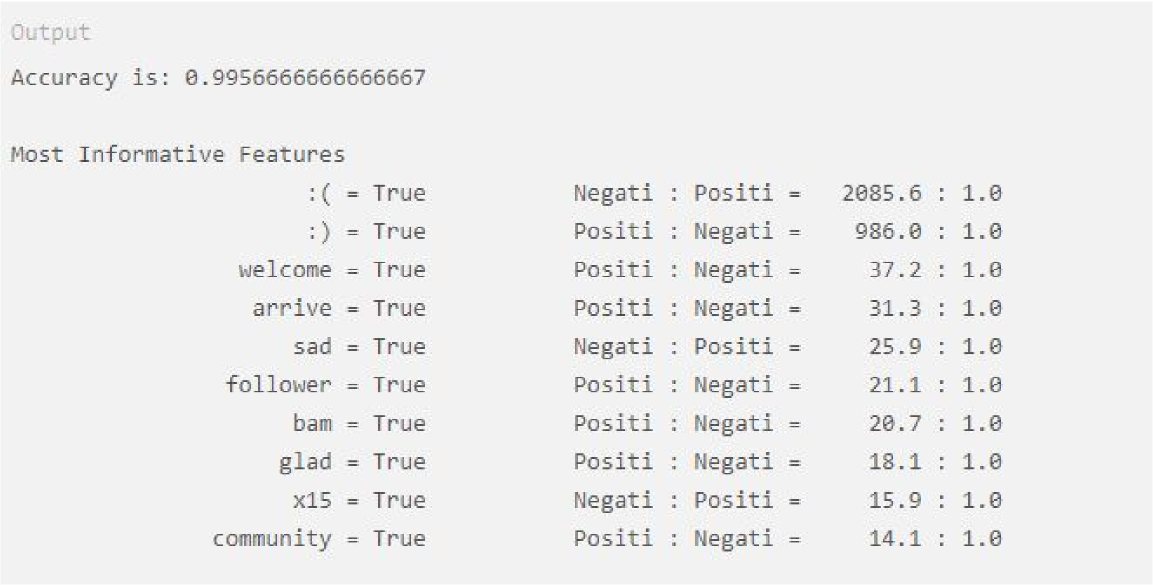
7.1 Accuracy

Machine learning model accuracy is the measurement used to determine which model is best at identifying relationships and patterns between variables in a dataset based on the input, or training, data. Formally, accuracy has the following definition:

Accuracy = $Number\ of\ correct\ predictions$ / $Total\ number\ of\ predictions$

Accuracy Obtained is 0.99

Chapter 8: Discussion of Results



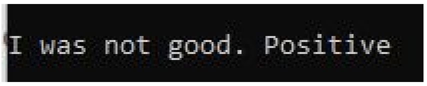
```
Output

Accuracy is: 0.9956666666666667

Most Informative Features
                      :( = True           Negati : Positi =   2085.6 : 1.0
                      :) = True           Positi : Negati =    986.0 : 1.0
                 welcome = True           Positi : Negati =     37.2 : 1.0
                  arrive = True           Positi : Negati =     31.3 : 1.0
                     sad = True           Negati : Positi =     25.9 : 1.0
                follower = True           Positi : Negati =     21.1 : 1.0
                     bam = True           Positi : Negati =     20.7 : 1.0
                    glad = True           Positi : Negati =     18.1 : 1.0
                     x15 = True           Negati : Positi =     15.9 : 1.0
               community = True           Positi : Negati =     14.1 : 1.0
```

The system gives out accuracy of 99 percent on unknown data. The most informative features given by the system are shown in the figure. It gives out the ratio of positive and negative emotions for each word in a text and compares their frequencies on the basis of which it is decided whether the text is positive or not.

As the training data and testing data are taken out from the same source and consist of similar types of data, the accuracy obtained on test data is very high. Whereas when a different data is provided to the model, it does not give a correct label. For double negation sentences, it does not give the correct label.



```
I was not good. Positive
```

The model focuses on individual words and for this particular sentence, it observes the word frequency of each word in positive and negative sentences. So, good has a high probability of appearing in a positive sentence so it declares the sentence as positive on the basis of frequency of good in the positive sentence.

Chapter 9: Contribution of Each Project Member

Teammates	**Contribution**
Yachana Aryal	All

Chapter 10: Code

10.1 Link to the GitHub repo of the code

The full code of the project "Sentiment Analysis" can be accessed through the public GitHub repository

from the given link:

https://github.com/yachana992/sentiment_analysis

Chapter 11: Conclusion and Future Extentions:

Sentiment analysis analyses a text and gives out the emotion depicted by that text. It takes text as an input from the user and gives out the sentiment or emotion that the particular text conveys. The sentiment can be positive or negative. The trained model gives 99 percent of accuracy on an unknown data. That means it gives out a correct output 99 percent of the time.

Whereas the system can be made better and more useful by introducing some other emotions like happy, sad, neutral, sarcastic etc. And based on the sentiment expressed by the tweets, it could recommend people the products they had positive sentiments toward. The system can be expanded in a way that gives out varieties of sentiments from the text. Furthermore, the system can be trained to analyse longer texts.