SENTIMENT ANALYSIS OF TWITTER DATASET USING NAÏVE BAYES ALGORITHM

J COMPONENT PROJECT REPORT

WINTER SEMESTER 2019-2020

Submitted by

SUGANDH AGGARWAL (17BCE0289)
VARUN KAUSHIK (17BCE0182)

in partial fulfillment for the award of the degree of

B. Tech

in

Computer Science and Engineering

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

Vellore-632014, Tamil Nadu, India

School of Computer Science and Engineering

May, 2020

Contents

Abstract

Sentiment analysis the process of interpretation and classification of emotions expressed in the form of text, especially to find out what is the writer's perception towards a particular topic, product, etc. is it positive, negative, or neutral.

These days the growth of social media is very rapid with the time population is becoming more familiar to the internet. So, the contribution of electronic media becoming very huge. Everyone expressing their attitudes towards any topic or product etc. using social media by comments tweets and reviews.

Sentiment Analysis is a term which is referred to collect information from a text by using Natural Language Processing and to make a decision by the information we extract and analyze the opinion of the writer and identifying how much the sentiment is positive or negative. Sentiment analysis concentrates to analyze people's reviews, attitudes, and emotions from written languages towards any product, particular topic, or any organization. Our main focus on different opinion classification techniques, performed on twitter data set which we will retrieve from twitter in real-time in a user interface and represent in the graphical form which will help in the analysis effectively.

Introduction

" Sentiment Analysis," collecting information from a text. For instance, movie reviews on IMDB are often positive or negative, product reviews on Flipkart. Similarly, people give opinions via tweets on Twitter. Some words have positive connotations (e.g.," like"), while some words tend to have negative connotations (e.g.," dislike"). And so, if someone tweet" I like you", you feel a positive sentiment. And if someone tweet is", I dislike you", you feel the negative sentiment.

Of course, you can't rely on individual words alone, as" I do not like you" isn't a positive sentiment, but you not bother about those circumstances. Also, some words, are neither positive nor negative emotions (e.g.," the").

As the internet is growing day by day, its area is becoming wider and wider. Social Media and Blogging platforms like Gig, Instagram, Facebook, Twitter play a big role in spreading news around the world. A topic comes in trending if more and more users are sharing their opinion and judgments on that trending topic, and it becomes a valuable source of online perception. These topics generally intended to spread awareness or to promote public figures, political campaigns during elections, product endorsements, and entertainment like movies, award shows. Large organizations and firms take advantage of people's feedback to enhance their products and services which further help in improving marketing strategies. One such example can be leaking the pictures of the upcoming smartphone to create a hype among peoples and market the product

before its release. At the same time, users also read reviews and tweets about a product before buying. What people thinking about the product who is already taking services from that product. Now the main problem arises is people always got positive as well negative feedbacks which creates doubt in the mind of the buyer whether to take that service or not. It is also impossible to read all reviews or tweets. Now, we are going to help them we will read all reviews for them and make a count and show how many positive, negative and neutral tweets about that product, organization, or service. Which helps in the decision making of the user.

Architecture diagram

- Pre-processing of training dataset is to be done before using the dataset. Pre-processing is required as the corpus is taken from book so, we have to remove whitespaces, punctuations marks, etc. After data cleaning tokenization is done. This tokenized data will be given to next stage.
- We have used the Naïve Bayes Classifier for classifying the sentiments into different categorize.
-

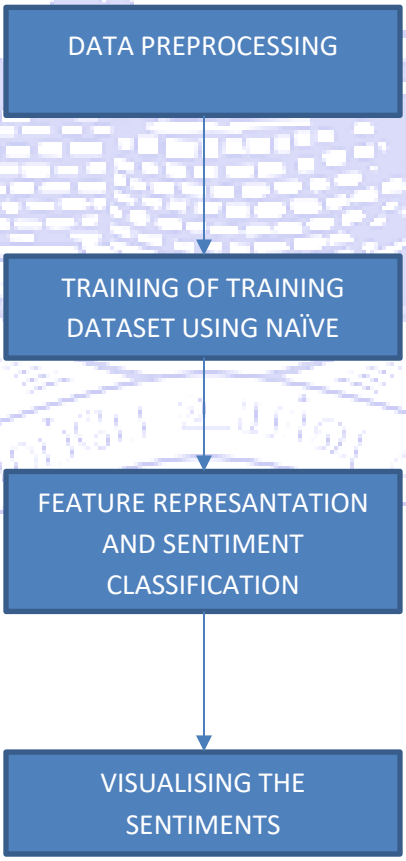

Fig-1

Background study

(Clavel and Callejas, 2015) The domain of sentiment analysis has seen an upsurge of curiosity with the fast expand of text data containing opinions, critics and suggestions on the web i.e. movie reviews, tweets, feedback etc. In this paper they establish and discuss the developing prospects for pass-disciplinary work that may expand individual advances. Sentiment or opinion detection methods used in human-agent interaction are certainly infrequent and, when they're employed, they are not exceptional from the ones utilized in opinion mining and therefore not designed for socio-affective interactions. They presented a comparative cutting-edge which analyses the sentiment-associated phenomena and the sentiment detection methods used in each community and makes an overview of the pursuits of socio-affective human-agent methods.

(Zhou, Wan and Xiao, 2016) Sentiment analysis of microblog texts has drawn countless attention in each the tutorial and industrial fields. However, lots of the current work only specializes in polarity classification. In this paper, they gift an opinion mining method for chinese microblogs known as CMiner. As a substitute of polarity classification, CMiner focuses on extra elaborate opinion mining duties - opinion goal extraction and opinion summarization. Novel algorithms are developed for the 2 duties and built-in into the end-to-end approach. A co-ranking algorithm is proposed to rank each the opinion targets and microblog sentences at the same time. Experimental results on a benchmark dataset show the effectiveness of our method and the algorithms

(Fang, Tan and Zhang, 2018) Western scholars first decided the sentiment tendency of words or phrases and quantified them as a measure of actual values, which can be used to determine the sentiment tendency of sentences and paragraphs. They analyzed the sentiment tendency of textual content via machine learning methods or ways established on polarity lexicon. In this paper, they recommend a multi-process sentiment evaluation approach with semantic fuzziness to solve the situation. The results exhibit that this hybrid sentiment analysis method can attain a good level of effectiveness.

(Shayaa et al, 2018) The data has sprung massively in quite a lot of fields over the last two decades, which has led to the birth of gigantic data. Furthermore, the influx of technology within the digital world has opened the doors for the development of gigantic information. In this regard, this paper provided a comprehensive systematic literature overview that objectives to discuss each technical side of OMSA (tactics and forms) and non-technical side in the form of application areas are discussed. Prior study suggests that high customers, as measured through the quantity of followers on Twitter, are usually celebrities and those who appeal to the keen interest of the mass media.

(Xu et al, 2019) Within the current sentiment analysis methods, distributed word is most likely used. However, distributed words most effectively consider the semantic meaning of phrase, but ignore the sentiment knowledge of the word. In this paper, an elevated word representation approach is proposed, which integrates the contribution of sentiment understanding into the common TF-IDF algorithm and generates weighted phrase vectors. Below the same conditions,

the proposed sentiment evaluation process is in comparison with the sentiment analysis approaches of RNN, CNN, LSTM, and NB. The experimental outcome show that the proposed sentiment analysis system has better precision, bear in mind, and F1 rating. The process is proved to be amazing with high accuracy on feedback.

(Yu et al, 2019) The normal sentiment analysis in most cases fascinated about the coarse-grained sentiment analysis on the document and sentence degree. The procedure would provide the emotional analysis established in most cases on the remarks. In an effort to obtain evaluation expertise in regards to the more than a few elements of products or services, the nice-grained matter Sentiment Unification (FG-TSU) model is proposed based on the advance of LDA (Latent Dirichlet Allocation) model. First of all, the themes are divided into nearby and global issues. Ultimately, they included the sentiment layer into LDA mannequin to receive the sentiment polarity of the whole review and specific points. They used hostel and mobile phone data for the above purpose.

(Lin, C et al 2016),In this paper they proposes a unique probabilistic modeling framework primarily based on Latent Dirichlet Allocation (LDA), referred to as joint sentiment/subject matter version (JST), which detects sentiment and topic simultaneously from textual content. Unlike different gadget getting to know approaches to sentiment category which often require classified corpora for classifier schooling, the proposed JST version is absolutely unsupervised. The version has been evaluated at the movie review dataset to categorize the overview sentiment polarity and minimal previous records have additionally been explored to further improve the sentiment type accuracy. Preliminary experiments have proven promising results performed by using JST.

Methodology

The proposed system consists of four modules - (1) Data preprocessing module: for preprocessing the data (2) Training of Training Dataset using Naïve Bayes Algorithm (3) Feature representation module and Sentiment classification using Naive base classifier (4) Visualizing the Sentiments in the form of Graphs or chart.

- Data preprocessing

    1. Extracting the Tweets Directly from Twitter using an API and tweepy libray.
    2. Retweets, which starts with “RT” are eliminated.
    3. User names preceded by ‘@- and external links are eliminated.
    4. hashtag ‘#, repeating letters, stop words, punctuation are removed from the tweets.
    5. We have converted all the tweets into lowercase.
    6. “Stemming” is done to reduce each word to its root word.

- Training of Training Dataset using Naïve Bayes Algorithm

  In this module we are training our training dataset using Naive Bayesian Classifier/ Naïve Bayes algorithm.

  Naive Bayesian Classifier:
  Naïve bayes is a prominent technique used in text classification. Even though it is a simple algorithm, the accuracy of the results given by this algorithm is high. Naïve bayes technique gives the results by calculating the chances of all the attributes of the dataset belonging to a particular class. Naïve bayes technique is framed on the principles if bayes theorem.
  The bayes theorem states that "the probability of event x given that event y has taken place is equal to the probability of event y given that event x has occurred multiplied by the probability of event x, divided by the probability of event y.

$$P(x|y) = \frac{P(y|x)P(x)}{P(y)}$$

P(T1,T2……..Tn/c) = P(T1/c).P(T2/c)……..P(Tn/c)

  Since the naïve bayes technique uses the linear combination of the terms to classify the data, it is known as linear classifier.

- Feature representation module and Sentiment classification using Naive base classifier:

  This module is responsible to extract features from preprocessed tweets. In this Project we have used naïve baiyes classifier as discussed above, So in this module we have classified the preprocessed/extracted tweets into the different categorize that is positive, negative or neutral. Consider the following three tweets:

  1.) Vistara is a good Airline. (Positive)
  2.) Vistara is not a bad Airline. (Neutral)
  3.) Vistara is a bad Airline. (Negative)

  So, in this way tweets are classified in this module.

- Visualizing the Sentiments in the form of Graphs or chart.In this module we are Visualizing the tweets into three categories that is Positive, Negative or Neutral using Pie-chart and Histogram. In this we are showing the percentage and the particular value of each sentiment, using these two charts.

Proposed model

- First, we have to take input from the user which he or she wants to analysis. Input is used as hashtag in API.Twitter API allows us to scrape tweets for any hash tags and store the tweets into a csv. So we are extracting the tweets of the keyword which the user gave.

  Ex -#vistara

- Now we preprocess the tweets which we have taken from twitter API.
- In preprocessing we are deleting the Stop words, punctuation, @username and retweeted tweets and we are cleaning the data. Smoothing is done so that we can find the probability of all the words in the tweets. There is various technique available for smoothing. But we have used Good Turing estimate as a smoothing technique. The Good-Turing estimate states that for any word that occur r times, we should re-estimate this frequency of occurrence.
- After preprocessing we will apply naïve Bayes with the training dataset which we have download from Kaggle after doing this step now we categories the result on the bases of weight for example if any tweet weight is negative we will count in negative counter if positive will count in positive after checking all tweets .Now we have count of all positive, negative tweets and neutral tweets. Now we have trained our training dataset, so now on the basis of that we are going to test our tweets/ dataset which we have created by extracting the tweets from twitter.

  Naïve Bayes Algorithm:

Naive Bayes is a simple technique for constructing classifiers: models that assign class labels to problem instances, represented as vectors of feature values, where the class labels are drawn from some finite set. There is not a single algorithm for training such classifiers, but a family of algorithms based on a common principle: all naive Bayes classifiers assume that the value of a particular feature is independent of the value of any other feature, given the class variable. For example, a fruit may be considered to be an apple if it is red, round, and about 10 cm in diameter. A naive Bayes classifier considers each of these features to contribute independently to the probability that this fruit is an apple, regardless of any possible correlations between the color, roundness, and diameter features.

Abstractly, naïve Bayes is a conditional probability model: given a problem instance to be classified, represented by a vector representing some *n* features (independent variables), it assigns to this instance probabilities

$$p(C_k \mid x_1, \ldots, x_n)$$

for each of K possible outcomes or *classes* C_k.[

Fig-2

The problem with the above formulation is that if the number of features *n* is large or if a feature can take on a large number of values, then basing such a model on probability tables is infeasible. We therefore reformulate the model to make it more tractable. Using Bayes' theorem, the conditional probability can be decomposed as

$$p(C_k \mid \mathbf{x}) = \frac{p(C_k)\, p(\mathbf{x} \mid C_k)}{p(\mathbf{x})}$$

Fig-3

Now the "naïve" conditional independence assumptions come into play: assume that all features in are mutually independent, conditional on the category .Under this assumption,

$$p(x_i \mid x_{i+1}, \ldots, x_n, C_k) = p(x_i \mid C_k)\,.$$

Thus, the joint model can be expressed as

$$\begin{aligned} p(C_k \mid x_1, \ldots, x_n) &\propto p(C_k, x_1, \ldots, x_n) \\ &= p(C_k)\, p(x_1 \mid C_k)\, p(x_2 \mid C_k)\, p(x_3 \mid C_k)\, \cdots \\ &= p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k)\,, \end{aligned}$$

Fig-4

- During testing we are going to test the extracted tweets and categorize them into the different-different sentiments that is Positive, negative and neutral.
- After this we have done effective visualization in which we have created a pie chart to get percentages of negatives positives and neutral tweets. We also draw a histogram to get visualize the count. Now we put all our graphs into t-kinter module for user interface to make it user friendly


TWEETS
INPUT HASHTAG
EXTRACT FROM API
PRE PPROCESSING
APPLYING NAIVE BAYES
TRAINING DATABASE
WEIGHT
POSITIVE
NEUTRAL
NEGATIVE
Process
Process
Process
POSITIVE COUNT++
NEUTRAL COUNT ++
NEGATIVE COUNT ++
Yes
TWEET >0
No
VISIUALIZATION
PIE CHART AND HISTOGRAM

Fig-5

Results and Discussion

- **Source code**

1.) Source Code for Input

```
import tkinter as tk
root= tk.Tk()
canvas1 = tk.Canvas(root, width = 400, height = 300)
canvas1.pack()
entry1 = tk.Entry (root)
x1=" "
canvas1.create_window(200, 140, window=entry1)
def Sentiment ():
    global x1
x1 = entry1.get()
label1 = tk.Label(root, text= x1)
canvas1.create_window(200, 230, window=label1)
 return x1
button1 = tk.Button(text='Generate sentiment', command=Sentiment)
canvas1.create_window(200, 180, window=button1)
root.mainloop()
l=x1
```

2.) Source code for extracting tweets

```
import tweepy
import csv
```

```
import pandas as pd

####input your credentials here

consumer_key='NQDhcsmQeHMV0nyng3a1mw1nM'

consumer_secret='7Kkz9e2LB79lqqup0rUXKZtiJObxH8MlPmBZyUbYWwSPYWHlYk'

access_token='2983419021-L6xGuDYvfy3Ae66nbxSOpUhAAQbCS8158zuHuWA'

access_secret='YxUBioJC2PWYQW3RDVl7ygIwJW90AOPX5BzMQoW7ZjlYi'

auth = tweepy.OAuthHandler(consumer_key, consumer_secret)

auth.set_access_token(access_token, access_secret)

api = tweepy.API(auth,wait_on_rate_limit=True)

csvFile = open('ua3.csv', 'a')

csvWriter = csv.writer(csvFile)

ua1=pd.DataFrame(columns=['created','tweet'])

#l=input("Enter the Keyword:")

for tweet in tweepy.Cursor(api.search,q='#'+l,count=3000,lang="en",since="2019-04-
03").items():

    ua1 = ua1.append({'created': tweet.created_at, 'tweet': tweet.text}, ignore_index=True)

    print (tweet.created_at, tweet.text)

    csvWriter.writerow([tweet.created_at, tweet.text.encode('utf-8')])
```

3.) Source code for Training the training Dataset.

```
import nltk

 training_set = nltk.classify.util.apply_features(extract_features, tweets)

 # Train the classifier Naive Bayes Classifier

 NBClassifier = nltk.NaiveBayesClassifier.train(training_set)
```

ua is a dataframe containing all the tweets of airline.

4.) Sample Code for Testing

ua1['sentiment'] = ua1['tweet'].apply(lambda tweet:

NBClassifier.classify(extract_features(getFeatureVector(processTweet2(tweet)))))

- **Screenshots**
  - Input

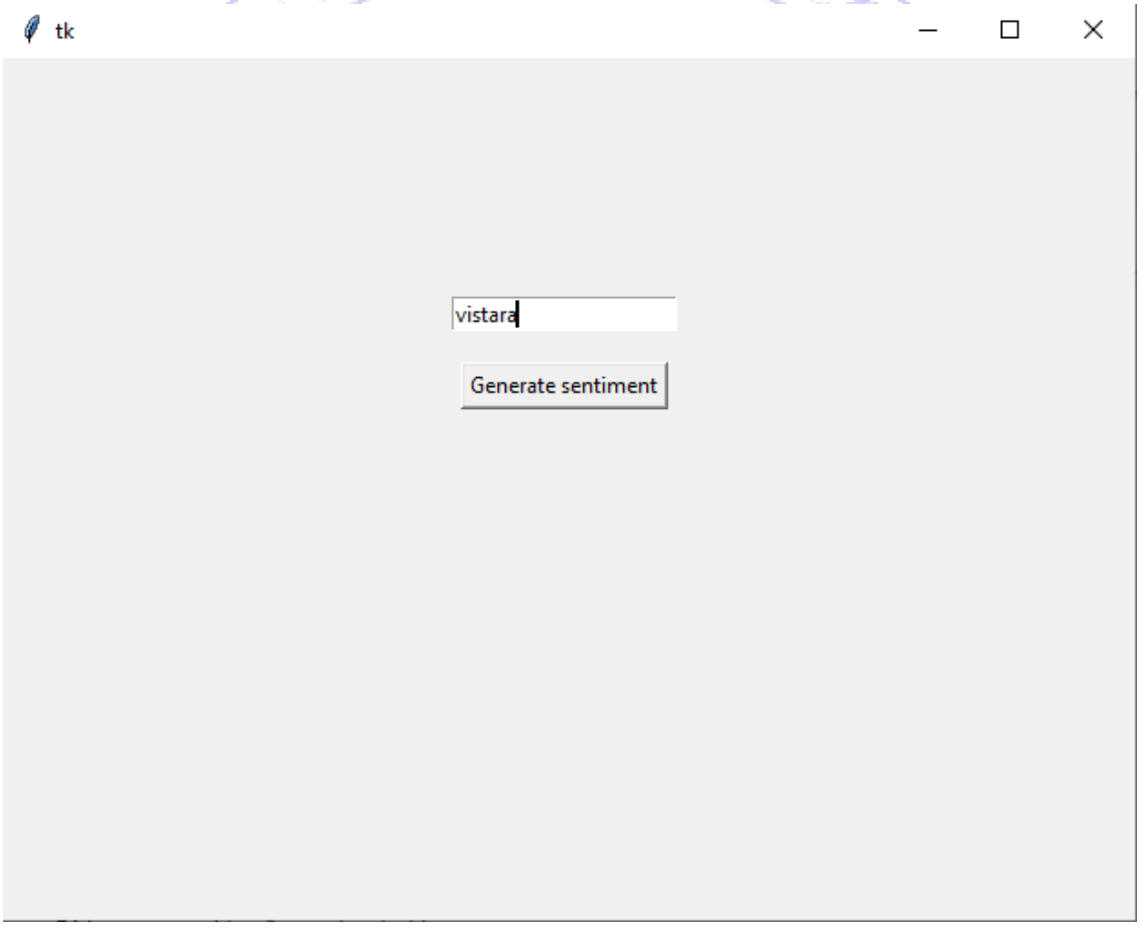

Fig-6

- Extracted Tweets

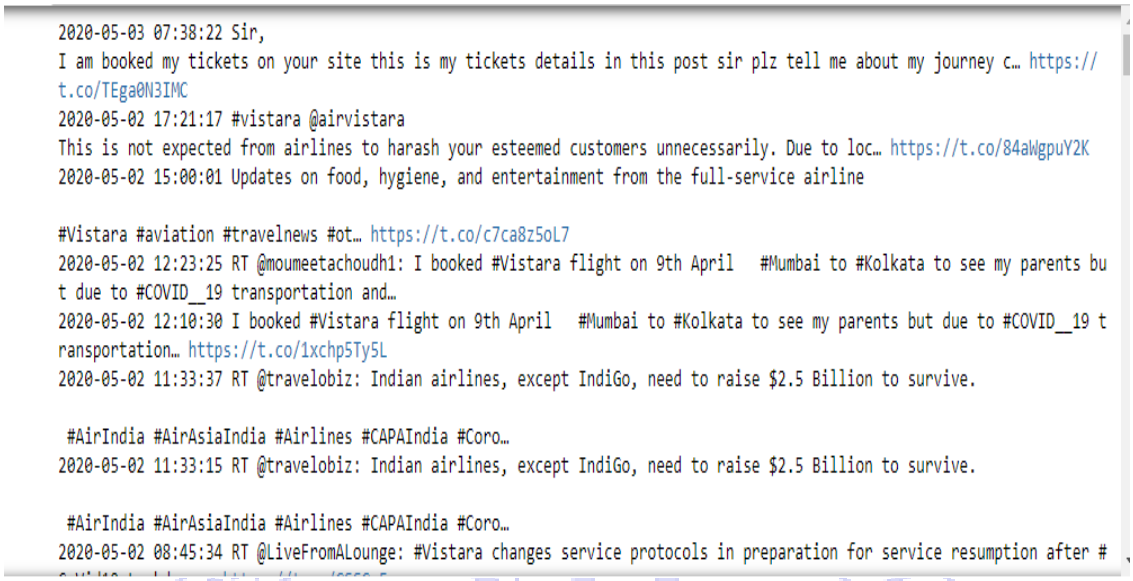

Fig-7

- Output

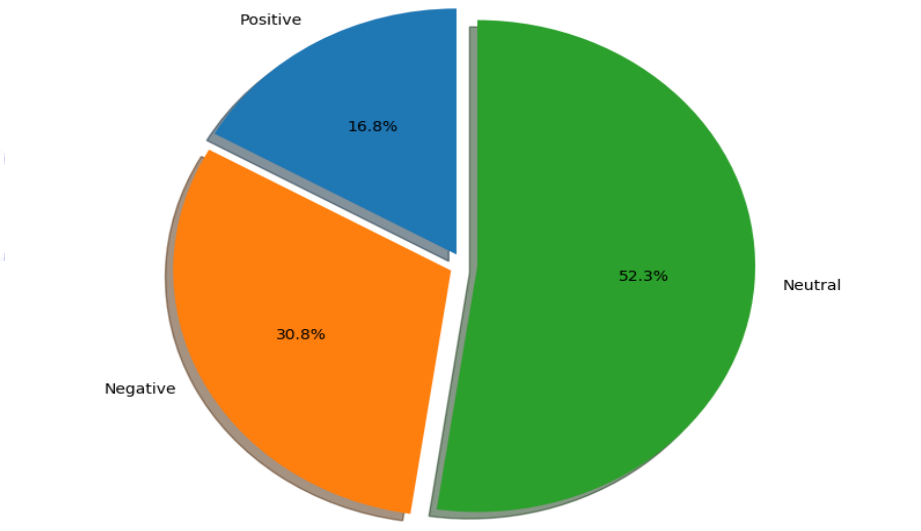

Fig-8 Pie Chart

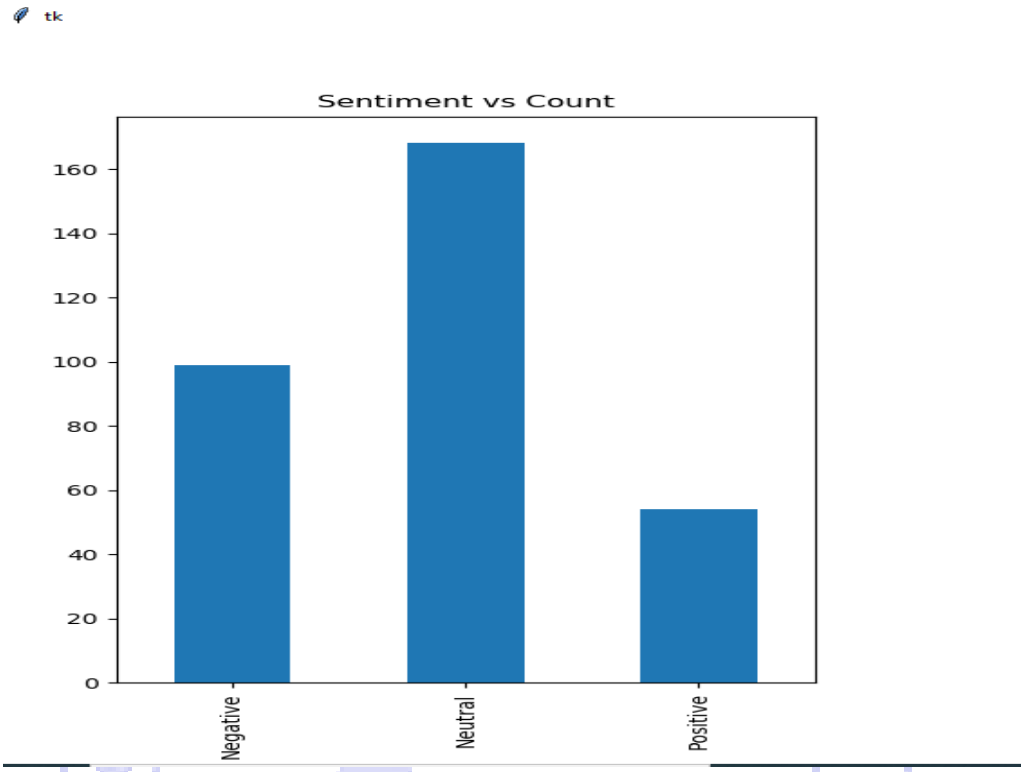

Fig-9 Histogram

So, we have successfully implemented the sentiment analysis of twitter dataset, because the output which we are getting is correct and having high Accuracy. We have used Naïve baiyes algorithm because it is less time consuming, can handle both continuous and discrete data, highly scalable and can work in less Training data.

Conclusion:

In this project, In the beginning, we were a bit doubtful about how NLTK and the Naïve Bayes algorithm will perform for Sentiment Analysis. In our experience, it works rather better for negative tweets. The problems come when the tweets are ironic, sarcastic has reference, or own difficult context.

We have successfully implemented the Naïve Bayse on the twitter data which we have imported from twitter API in real-time. We also successfully develop UI for input and producing the results in the form of pie chart and histogram for a better understanding of the proportion of negative-positive or neutral tweets about the hashtag input by the user.

References:

- Clavel, C. and Callejas, Z., 2015. Sentiment analysis: from opinion mining to human-agent interaction. IEEE Transactions on affective computing, 7(1), pp.74- 93.

- Fang, Y., Tan, H. and Zhang, J., 2018. Multi-strategy sentiment analysis of consumer reviews based on semantic fuzziness. Ieee Access, 6,pp.20625-20631.

- Lin, C. and He, Y., 2016, November. Joint sentiment/topic model for sentiment analysis. In *Proceedings of the 18th ACM conference on Information and knowledge management* (pp. 375-384).

- Shayaa, S.,Jaafar, N.I., Bahri, S., Sulaiman, A., Wai, P.S., Chung, Y.W., Piprani, A.Z. and Al-Garadi, M.A., 2018. Sentiment analysis of big data: Methods, applications, and open challenges. IEEE Access, 6, pp.37807-37827.

- Xu, G., Meng, Y., Qiu, X., Yu, Z. and Wu, X., 2019. Sentiment analysis of comment texts based on BiLSTM. Ieee Access, 7, pp.51522-51532.

- Yu, L., Wang, L., Liu, D. and Liu, Y., 2019. Research on Intelligence Computing Models of Fine-Grained Opinion Mining in Online Reviews. IEEE Access, 7, pp.116900-116910.

- Zhou, X., Wan, X. and Xiao, J., 2016. Cminer: opinion extraction and summarization for chinese microblogs. IEEE Transactions on Knowledge and Data Engineering, 28(7), pp.1650-1663.