SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE

A PROJECT REPORT ON

"CLASSIFY TWEETS INTO POSITIVE AND NEGATIVE TWEETS"

SUBMITTED TOWARDS THE PARTIAL FULFILLMENT

OF REQUIREMENT OF

Data Science and Big Data Analytics Laboratory

IN THIRD YEAR COMPUTER ENGINEERING

BY

Hulule Siddhant Gahininath [3153]
Gangurde Tejas Ramesh [3141]
Golhar Pankaj Gorakshanath[3146]

(T. E.COMPUTER ENGINEERING)

UNDER THE GUIDANCE OF

DR. R.G.Tambe

DURING THE ACADEMIC YEAR 2023-2024 (Sem-VI)

Department of Computer Engineering
Amrutvahini College of Engineering,
Sangamner- 422 608

Amrutvahini College of Engineering,Sangamner

CERTIFICATE

This is to certify that the project entitled

Hulule Siddhant Gahininath [3153]
Gangurde Tejas Ramesh [3141]
Golhar Pankaj Gorakshanath[3146]

T. E. (A) COMPUTER ENGINEERING

have successfully completed the work associated with Data science And Big Data Analytics Laboratory titled as

"CLASSIFY TWEETS INTO POSITIVE AND NEGATIVE TWEETS"

and has submitted the work book associated under my supervision, in the partial fulfillment of Third Year Bachelor of Engineering (2019 course) of Savitribai Phule Pune University.

Dr. R. G. Tambe

Guide

ACKNOWLEDGEMENT

With deep sense of gratitude we would like to thank all the people who have lit our path with their kind guidance. We are very grateful to these intellectuals who did their best to help during our project work

It is our proud privilege to express a deep sense of gratitude to Dr. M. A. Venkatesh Principal of Amrutvahini College of Engineering, Sangamner, for his comments and kind permission to complete this project. We remain indebted to Dr. S. K. Sonkar, H.O.D. Computer Engineering Department for his timely suggestion and valuable guidance.

The special gratitude goes to Dr. D. R. Patil for their excellent and precious guidance in completion of this work. We thanks to all the colleagues for their appreciable help for our working project. With various industry owners or lab technicians to help, it has been our endeavour throughout our work to cover the entire project work

We are also thankful to our parents who provided their wishful support for our project completion successfully, and lastly we thank our all friends and the people who are directly or indirectly related to our project work.

Hulule Siddhant Gahininath [3153]
Gangurde Tejas Ramesh [3141]
Golhar Pankaj Gorakshanath[3146]
T.E. Computer Engineering

TABLE OF CONTENTS

ABSTRACT

This project addresses the problem of sentiment analysis in twitter; that is classifying tweets according to the sentiment expressed in them: positive, negative or neutral. Twitter is an online micro-blogging and social-networking platform which allows users to write short status updates of maximum length 140 characters. It is a rapidly expanding service with over 200 million registered users out of which 100 million are active users and half of them log on twitter on a daily basis - generating nearly 250 million tweets per day. Due to this large amount of usage we hope to achieve a reflection of public sentiment by analyzing the sentiments expressed in the tweets. Analyzing the public sentiment is important for many applications such as firms trying to find out the response of their products in the market, predicting political elections and predicting socioeconomic phenomena like stock exchange. The aim of this project is to develop a functional classifier for accurate and automatic sentiment classification of an unknown tweet stream.

INTRODUCTION

This project of analyzing sentiments of tweets comes under the domain of "Pattern Classification" and "Data Mining". Both of these terms are very closely related and intertwined, and they can be formally defined as the process of discovering "useful" patterns in large set of data, either automatically (unsupervised) or semi-automatically (supervised). The project would heavily rely on techniques of "Natural Language Processing" in extracting significant patterns and features from the large data set of tweets and on "Machine Learning" techniques for accurately classifying individual unlabeled data samples (tweets) according to whichever pattern model best describes them. The features that can be used for modeling patterns and classification can be divided into two main groups: formal language based and informal blogging based. Language based features are those that deal with formal linguistics and include prior sentiment polarity of individual words and phrases, and parts of speech tagging of the sentence. Prior sentiment polarity means that some words and phrases have a natural innate tendency for expressing particular and specific sentiments in general. For example the word "excellent" has a strong positive connotation while the word "evil" possesses a strong negative connotation. So whenever a word with positive connotation is used in a sentence, chances are that the entire sentence would be expressing a positive sentiment. Parts of Speech tagging, on the other hand, is a syntactical approach to the problem. It means to automatically identify which part of speech each individual word of a sentence belong.

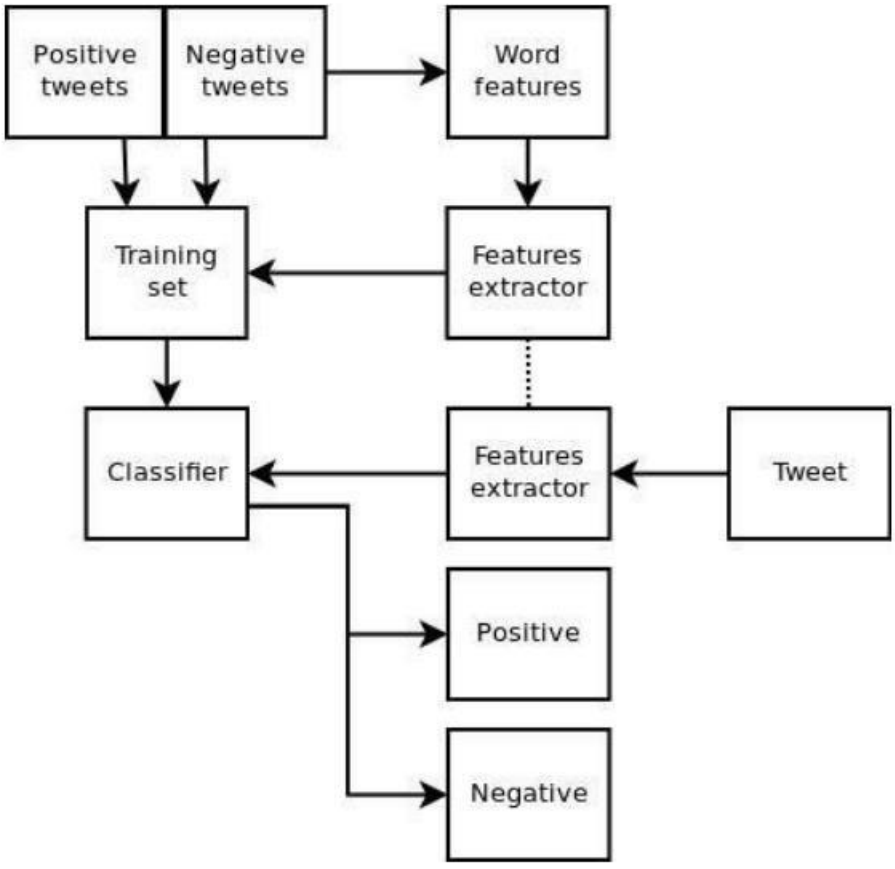

Fig : Twitter Sentiment Analysis

PROPOSED SYSTEM

Problem Statement: Use the following dataset and classify tweets into positive and negative tweets. https://www.kaggle.com/ruchi798/data-science-tweets.

Hardware Requirement: A PC with Windows/Linux OS Processor with 1.7-2.4gHz speed Minimum of 8gb RAM 2gb Graphic card

Software Requirement: Text Editor (VS-code/WebStorm) Anaconda distribution package (PyCharm Editor) Python libraries

Libraries used:

A. Pandas

B. Numpy

C. Scikit-learn

D. Seaborn

E. Matplotlib **Requirements:**

Python 3.6

- What is Tweets classification?

"Tweet Classification into Positive and Negative Tweets" refers to the task of analysing tweets posted on the social media platform Twitter and categorizing them based on their sentiment. The goal is to develop algorithms and methodologies that can automatically determine whether a tweet expresses positive or negative sentiment. This task is essential for various applications, including understanding public opinion, monitoring brand sentiment, and analysing societal trends. By classifying tweets into positive and negative categories, organizations and researchers can gain valuable insights into the overall sentiment of Twitter users and respond accordingly.

PROJECT CODE AND OUTPUT

In [1]:
```
import pandas as pd
df = pd.read_csv('data_visualization.csv)
```

```
/var/folders/v3/cy6m4_ms66s6d1zfvqb7vfw00000gp/T/ipykernel_7140/1333053867.py:2: DtypeWarning: Columns (22,24) have mixed types. Specify dtype option on import or set low_memory=False.
  df = pd.read_csv('data_visualization.csv')
```

In [2]:
```
df.info()
```

```
<class                                                  ame.DataFrame'>
        eIndex: 33590                                   s, 0 to 33589   36
        columns (total                                  ns):
 #      Column                                          Non-Null Count Dtype
------  ------                                          --------------------
 0 1    id                                              33590 non-null int64 non-
 2      conversation_id 33590                           null int64
 3 4    created_at                                      33590 non-null object
 5 6    date time                                       33590 non-null object
 7      timezone                                        33590 non-null object
 8      user_id                                         33590 non-null int64
 9      username                                        33590 non-null int64
 10     name                                            33590 non-null object
 11 12  place                                           33590 non-null object
 13     tweet                                           85 non-null
 14 15  language                                        33590 non-null object   ject
 16 17  mentions                                        33590 non-null object
 18 19  urls photos                                     33590 non-null object
 20     replies_count                                   33590 non-null object
 21     retweets_count                                  33590 non-null int64
 22     likes_count                                     33590 non-null int64
 23     hashtags                                        33590 non-null int64
 24     cashtags link                                   33590 non-null object
 25     retweet                                         33590 non-null object
 26                quote_url                            33590 non-null object
 27                video                                33590 non-null bool
 28              thumbnail                                1241 non-null
 29             near geo                                 33590 non-null   object
 30               source                                  9473 non-null   int64
 31            user_rt_id                                    0 non-null   object
 32              user_rt                                     0 non-null   float64
 33           retweet_id                                     0 non-null   float64
 34             reply_to                                     0 non-null   float64
 35         retweet_date                                     0 non-null   float64
dtypes:          translate                                  0 non-null   float64
9.0+             trans_src                                  0 non-null   float64
                 trans_dest                             33590  non-null   float64
                   bool(1),                                 0 non-null   object
                  float64(10),                              0 non-null   float64
                 MB                                         0 non-null   float64
                                                            0 non-null   float64
                                                          int64(8),                  ') memory usage:
```

In [3]:

Out[3]:
```
df['tweet'][10]
```

'We are pleased to invite you to the EDHEC DataViz Challenge grand final for a virtual exchange with all Top 10 finalists to see how data visualization creates impact and can bring out compelling stories in support of @UNIC EF's mission. https://t.co/Vbj9B48VjV'

In [4]:

```
import nltk nltk.download('vader_lexicon')
from nltk.sentiment.vader import SentimentIntensityAnalyzer sid =
SentimentIntensityAnalyzer()
import re
import pandas as pd
import nltk
nltk.download('words')
words = set(nltk.corpus.words.words())
```

```
[nltk_data] Downloading package vader_lexicon to
[nltk_data]             /Users/Smiti/nltk_data...
[nltk_data]         Package vader_lexicon is already up-to-date! [nltk_data]
Downloading package words to /Users/Smiti/nltk_data... [nltk_data]
                    Package words is already up-to-date!
```

In [5]:
```
sentence = df['tweet'][0] sid.polarity_scores(sentence)['compound']
```

Out[5]:
```
0.7089
```

n [6]:
```
def cleaner(tweet):
    tweet = re.sub("@[A-Za-z0-9]+","",tweet) #Remove @ sign
    tweet = re.sub(r"(?:\@|http?\://|https?\://|www)\S+", "",tweet) #Remove http links
    tweet = " ".join(tweet.split())
    tweet = tweet.replace("#", "").replace("_", " ") #Remove hashtag sign but keep the text
    tweet = " ".join(w for w in nltk.wordpunct_tokenize(tweet) if w.lower() in words or not w.isalpha())
    return tweet
```

```
df['tweet_clean'] = df['tweet'].apply(cleaner)
```

In [7]:
```
word_dict ={'manipulate':-1,'manipulative':-1,'jamescharlesiscancelled':-1,'j ameschar lesisoverparty':-1,

'pedophile':-1,'pedo':-1,'cancel':-1,'cancelled':-1,'cancel culture':0.4,'teamtati':-1,'teamjames':1,'teamjames
import nltk nltk.download('vader_lexicon')
from nltk.sentiment.vader import SentimentIntensityAnalyzer sid =
SentimentIntensityAnalyzer() sid.lexicon.update(word_dict)
list1 = []
for i in df['tweet_clean']: list1.append((sid.polarity_scores(str(i)))['compound'])
```

In [9]:

Out[9]:
```
[nltk_data] Downloading package vader_lexicon to
[nltk_data]           /Users/Smiti/nltk_data...
[nltk_data]         Package vader_lexicon is already up-to-date!
```

In [8]:
```
df['sentiment'] = pd.Series(list1)
def sentiment_category(sentiment):
    label = ''
    if(sentiment>0):  label =
        'positive'
    elif(sentiment == 0):
        label = 'neutral'
    else:
        label = 'negative'
    return(label)
df['sentiment_category'] = df['sentiment'].apply(sentiment_category)
```

	tweet	date	id	sentiment	sentiment_category
0	Take your storytelling to the next level using...	2021-06-20	1406335989484822531	0.7089	positive
1	Choosing Fonts for Your Data Visualization | b...	2021-06-19	1406292636789526537	0.0000	neutral
2	This data visualization shows where our greate...	2021-06-19	1406082288035811330	0.0000	neutral
3	Looking for examples of stellar charts made so...	2021-06-18	1405948260796100610	0.4019	positive

[10]:
```
neg = df[df['sentiment_category']=='negative'] neg =
neg.groupby(['date'],as_index=False).count() pos =
df[df['sentiment_category']=='positive'] pos =
pos.groupby(['date'],as_index=False).count() pos =
pos[['date','id']]
neg = neg[['date','id']]
```

```
import plotly.graph_objs as go fig =
go.Figure()
for col in pos.columns: fig.add_trace(go.Scatter(x=pos['date'],
      y=pos['id'],
                                          name = col,
                                          mode = 'markers+lines',
                                          line=dict(shape='linear'),
                                          connectgaps=True, line_color='green'
                                         )
                        )
for col in neg.columns: fig.add_trace(go.Scatter(x=neg['date'],
      y=neg['id'],
                                         name = col,
                                         mode = 'markers+lines',
                                         line=dict(shape='linear'),
                                         connectgaps=True,
                                         line_color='red'
                                         )
                        )
fig.show()
```

In [11]:

CONCLUSION

The task of sentiment analysis, especially in the domain of micro -blogging, is still in the developing stage and far from complete. So we propose a couple of ideas which we feel are worth exploring in the future and may result in further improved performance. Right now we have worked with only the very simplest unigram models; we can improve those models by adding extra information like closeness of the word with a negation word. We could specify a window prior to the word (a window could for example be of 2 or 3 words) under consideration and the effect of negation may be incorporated into the model if it lies within that window. The closer the negation word is to the unigram word whose prior polarity is to be calculated, the more it should affect the polarity. For example if the negation is right next to the word, it may simply reverse the polarity of that word and farther the negation is from the word the more minimized ifs effect should be.