Final Report

On

**SENTIMENTAL ANALYSIS**



School of Computer Science and Engineering

Lovely Professional University.

Phagwara, Punjab (India).

INT 404 – ARTIFICIAL INTELLIGENCE

Submitted to:

Mr. Sagar Pandey sir.

Submitted By:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Roll no** | **Registration No** | **Name of the Student** | **Marks** | **Signature** |
| **02** | **11811276** | **N. Srikanth** |  |  |
| **25** | **11810917** | **P. Santosh Kumar** |  |  |

ABSTRACT

Senimental Analysis is all about analyzing the review or the opinion given by a customer or any person in the form of online tweets or online article .It is the process of computationally identifying and categorizing opinions from piece of text and determine whether the writer’s attitude towards the particular topic or the product is positive, negative or neutral.

Now a days this is playing a key role in many business organizations to imporove their products and to satisfy the needs of their customers more effectively. As of now, when compared to previous years data is generating at a rapid rate , so it became very difficult for a human to analyse all the reviews given by the customer from such huge data , this kind of projects makes the things easier for the companies or movie directors , for whoever want to estimate a product review.

This project is used in various sectors to categorize the audience opinion to their work/product.

Related work:

Natural Language Processing (NLP) is a hotbed of research in data science and one of the most common applications of NLP is sentiment analysis. From opinion polls to creating entire marketing strategies, this domain has completely reshaped the way businesses work, which is why this is an area every data scientist must be familiar with.

Thousands of text documents can be processed for sentiment (and other features including named entities, topics, themes, etc.) in seconds, compared to the hours it would take a team of people to manually complete the same task.

We will do so by following a sequence of steps needed to solve a general sentiment analysis problem. We will start with preprocessing and cleaning of the raw text of the tweets. Then we will explore the cleaned text and try to get some intuition about the context of the tweets. After that, we will extract numerical features from the data and finally use these feature sets to train models and identify the sentiments of the tweets.

**Implementation of code:**

We have Implemented our project in two ways:

1. Sentimental Analysis using tweepy
2. Sentimental Analysis of online articles.

**Santosh:**

1. Using tweepy

import tweepy  
import sys  
from textblob import TextBlob  
import matplotlib.pyplot as plt  
import re  
import pandas as pd  
import numpy as np  
  
*#defining the keys ---from twitter api*Consumer\_Key = "qhfac9zsMmUCa8GkMLOYM21hg"  
Consumer\_Secret = "PNgSjpiu4rEHO2FkIri4mseCksGvxGFFbb0LluwrBUOhvtsGiI"  
Access\_token = "472099927-B03YMYZ8R88OTnQGIxvEN45mfrg2rNaUJNzP222K"  
Access\_token\_Secret = "oE8fAROsHENZzCNfozla6z6t6elgllhU0eeRBBLi2Iovq"  
  
*#creating the authentication*auth = tweepy.OAuthHandler(consumer\_key= Consumer\_Key,consumer\_secret=Consumer\_Secret)  
auth.set\_access\_token(Access\_token,Access\_token\_Secret)  
api = tweepy.API(auth,wait\_on\_rate\_limit=True)  
*#asking user to enter the keyword and number of tweets to be analysed.*search\_word = input("Enter the Keyword/hashtag to be searched:")  
num\_of\_tweets = int(input("Enter the Number of tweets to be analysed:"))

tweets = tweepy.Cursor(api.search , q=search\_word , language =**"English"**).items(num\_of\_tweets)  
**for** tweet **in** tweets:  
 analysis = TextBlob(tweet.text)  
 polarity = analysis.sentiment.polarity  
 *#Creating the Data Frame of Tweets* df = pd.DataFrame([tweet.text **for** tweet **in** tweets] , columns= [**'Tweets'**])  
 *#print(analysis)* print(df.head())  
  
*#cleaning the tweets:***def** clean\_tweet(txt):  
 txt = re.sub(**'@[A-Za-z0–9]+'**, **''**, txt) *#Removing @mentions* txt = re.sub(**'#'**, **''**, txt) *# Removing '#' hash tag* txt = re.sub(**'RT[\s]+'**, **''**, txt) *# Removing RT* txt = re.sub(**'https?:\/\/\S+'**, **''**, txt) *# Removing hyperlink* **return** txt

df[**'Tweets'**] = df[**'Tweets'**].apply(clean\_tweet)  
  
*#printing the cleaned tweets:*print(df.head())  
  
*#creating function to get the subjectivity***def** subjectivity(txt):  
 **return** TextBlob(txt).sentiment.subjectivity  
*#creating a function to get the polarity***def** t\_polarity(txt):  
 **return** TextBlob(txt).sentiment.polarity  
  
*#now creating two new columns in the created dataframe to store polarity and subjecttivity.*df[**'subjectivity'**] = df[**'Tweets'**].apply(subjectivity)  
df[**'polarity'**] = df[**'Tweets'**].apply(t\_polarity)  
*#displaying the new df with both subjectiviy and polarity*print(df.head())

*# Plotting*plt.figure(figsize=(10, 10))  
**for** i **in** range(0, df.shape[0]):  
 plt.scatter(df[**"polarity"**][i], df[**"subjectivity"**][i], color=**'Blue'**) *# plt.scatter(x,y,color)*plt.suptitle(**"\tHow People are reacting on "**+ search\_word + **" by analysing "**+ str(num\_of\_tweets) +**" Tweets ."**)  
plt.title(**'Sentiment Analysis'**)  
plt.xlabel(**'polarity'**)  
plt.ylabel(**'subjectivity'**)  
plt.show()

1. Sentimental Analysis of article

!pip install article

!pip install newspaper3k

#importing the libraries

from textblob import TextBlob

from newspaper import Article

import nltk

#getting the url

url = 'https://www.who.int/emergencies/diseases/novel-coronavirus-2019/global-research-on-novel-coronavirus-2019-ncov'

article = Article(url)

#on performing nlp on the given url

article.download()

article.parse()

nltk.download('punkt')

article.nlp()

text = article.summary

print(text)

art = TextBlob(text)

polarity = art.sentiment.polarity

#print(polarity)

if (polarity == 0):

  print("The text extracted from the given url is Neutral.")

elif (polarity > 0):

  print("The text extracted from the given url is Positive.")

elif(polarity <0):

  print("The text extracted from the given url is Negative.")

**Srikanth:**

*we have completed the cleaning and assigning the polarity task***def** getAnalysis(score):  
 **if** score < 0:  
 **return 'Negative'  
 elif** score == 0:  
 **return 'Neutral'  
 else**:  
 **return 'Positive'**df[**'Analysis'**] = df[**'polarity'**].apply(getAnalysis)  
*#printing positive tweets*print(**'Printing positive tweets:\n'**)  
j=1  
sortedD\_Frame = df.sort\_values(by=[**'polarity'**]) *#Sort the tweets***for** i **in** range(0, sortedD\_Frame.shape[0] ):  
 **if**( sortedD\_Frame[**'Analysis'**][i] == **'Positive'**):  
 print(str(j) + **') '**+ sortedD\_Frame[**'Tweets'**][i])  
 print()  
 j= j+1  
  
*# Printing negative tweets*print(**'Printing negative tweets:\n'**)  
j=1  
sortedDF = df.sort\_values(by=[**'polarity'**],ascending=**False**) *#Sort the tweets***for** i **in** range(0, sortedD\_Frame.shape[0] ):  
 **if**( sortedD\_Frame[**'Analysis'**][i] == **'Negative'**):  
 print(str(j) + **') '**+sortedD\_Frame[**'Tweets'**][i])  
 print()  
 j=j+1  
*# Printing neutral tweets*print(**'Printing negative tweets:\n'**)  
j=1  
sortedDF = df.sort\_values(by=[**'polarity'**],ascending=**False**) *#Sort the tweets***for** i **in** range(0, sortedD\_Frame.shape[0] ):  
 **if**( sortedD\_Frame[**'Analysis'**][i] == **'Neutral'**):  
 print(str(j) + **') '**+sortedD\_Frame[**'Tweets'**][i])  
 print()  
 j=j+1

*#now creating two new columns in the created dataframe to store polarity and subjecttivity.*df[**'subjectivity'**] = df[**'Tweets'**].apply(subjectivity)  
df[**'polarity'**] = df[**'Tweets'**].apply(t\_polarity)  
*#displaying the new df with both subjectiviy and polarity*print(df.head())

*# Print the percentage of positive tweets*ptweets = df[df.Analysis == **'Positive'**]  
ptweets = ptweets[**'Tweets'**]  
ptweets  
round( (ptweets.shape[0] / df.shape[0]) \* 100 , 1)  
*# Print the percentage of negative tweets*ntweets = df[df.Analysis == **'Negative'**]  
ntweets = ntweets[**'Tweets'**]  
ntweets  
round( (ntweets.shape[0] / df.shape[0]) \* 100, 1)  
*# Show the value counts*df[**'Analysis'**].value\_counts()  
*# Plotting and visualizing the counts*plt.title(**'Sentiment Analysis'**)  
plt.xlabel(**'Sentiment'**)  
plt.ylabel(**'Counts'**)  
df[**'Analysis'**].value\_counts().plot(kind = **'bar'**)  
plt.show()

**Result:**

1. Output format of Sentimental Analysis using tweepy in python.

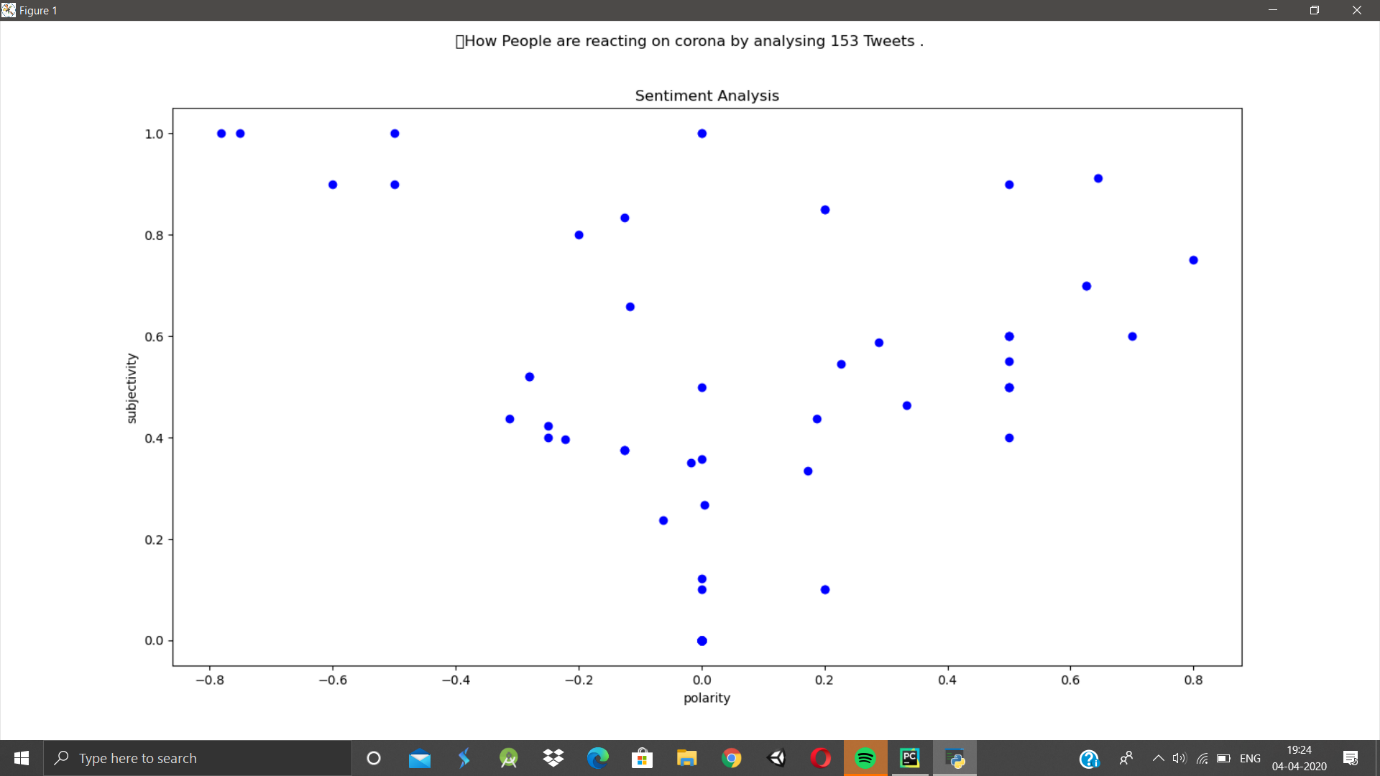
The output will display the scatter plot and it will print all the positive,negative ,neutral tweets with scatter plot and bar graph accordingly for the corresponding tweets analysed.

Eg: After compilation it will asks the user to enter the Search word or the trending hashtag and the number of tweets to be analysed.

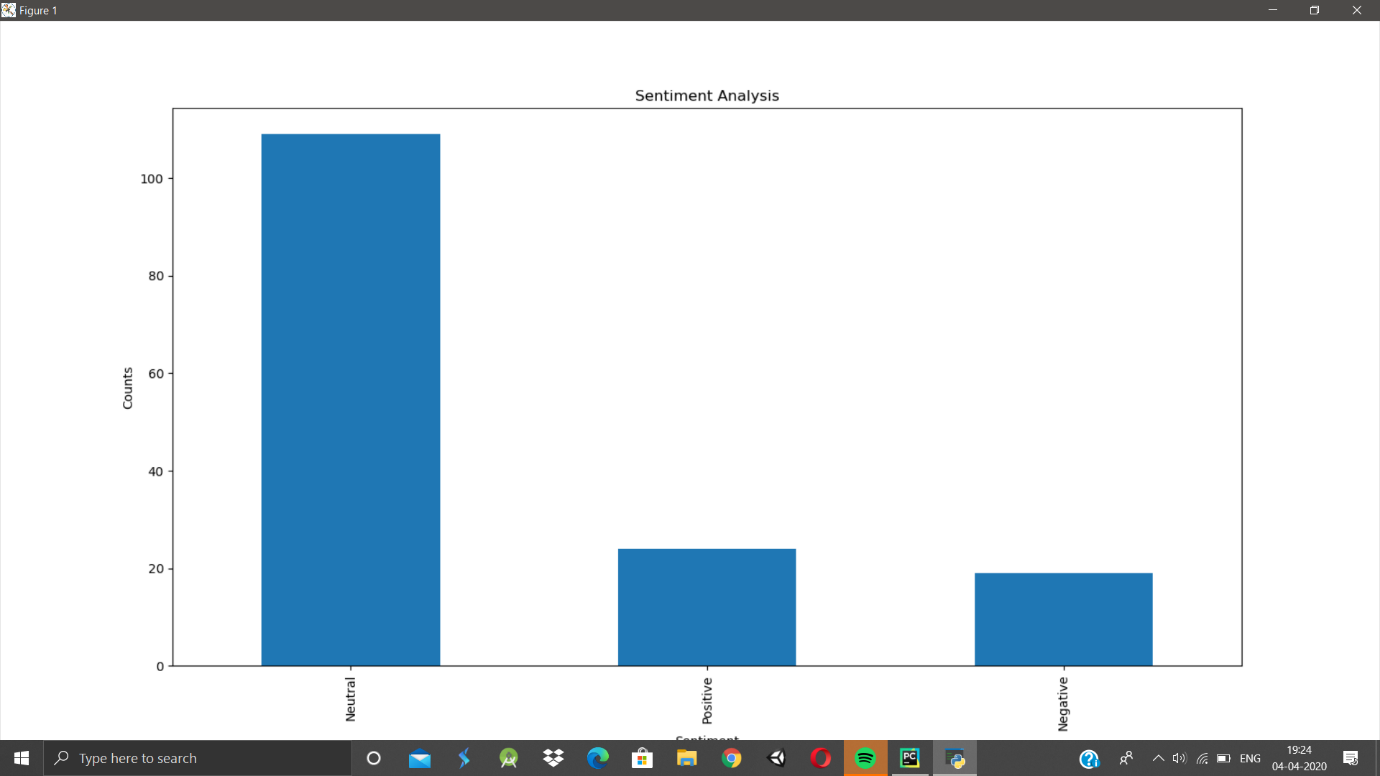
Then after analysing the given number of tweets it will display the bar graph and scatter plot showing the polarity and subjectivity of the tweets & the tweets classified as Positive , negative and neutral.

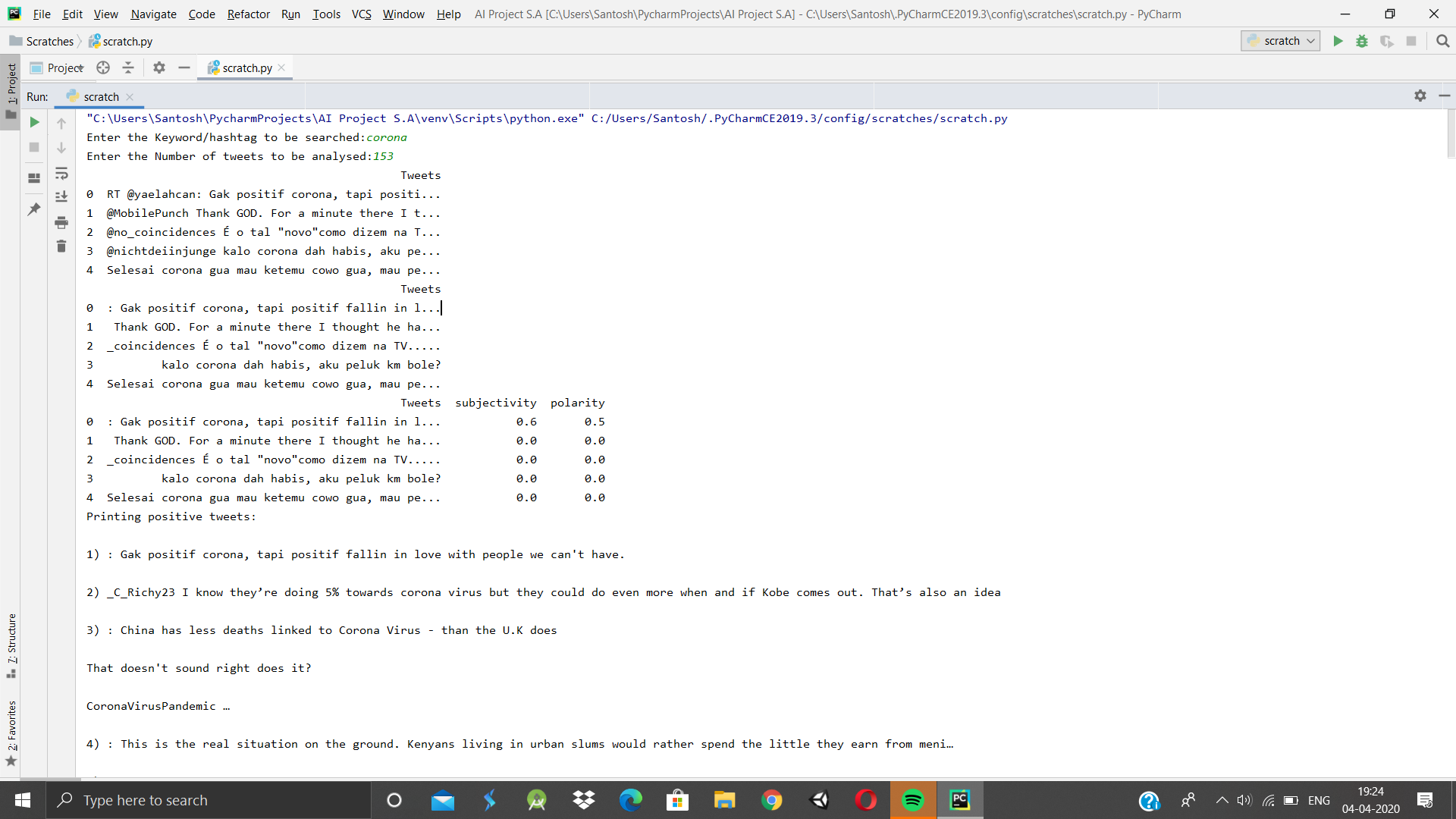
**Screen Shots:**

1. As corona virus is trending now we will give it as input and by analysing 153 tweets the output is

Scatter Plot of Corona virus tweets

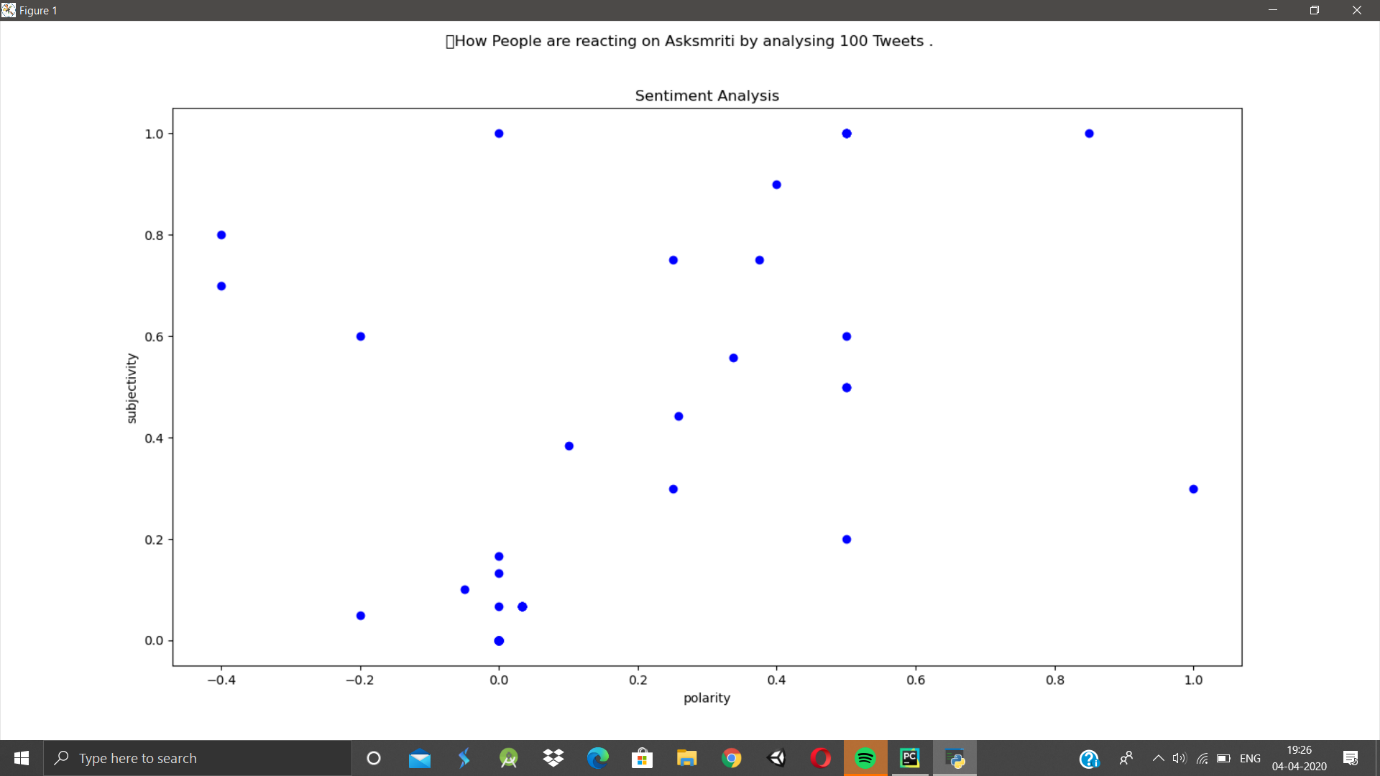
Always do not expect polarity should be negative and the plot should be extremely left here it did not happened as it extracts tweets randomly there might be some tweets in favour of the content.



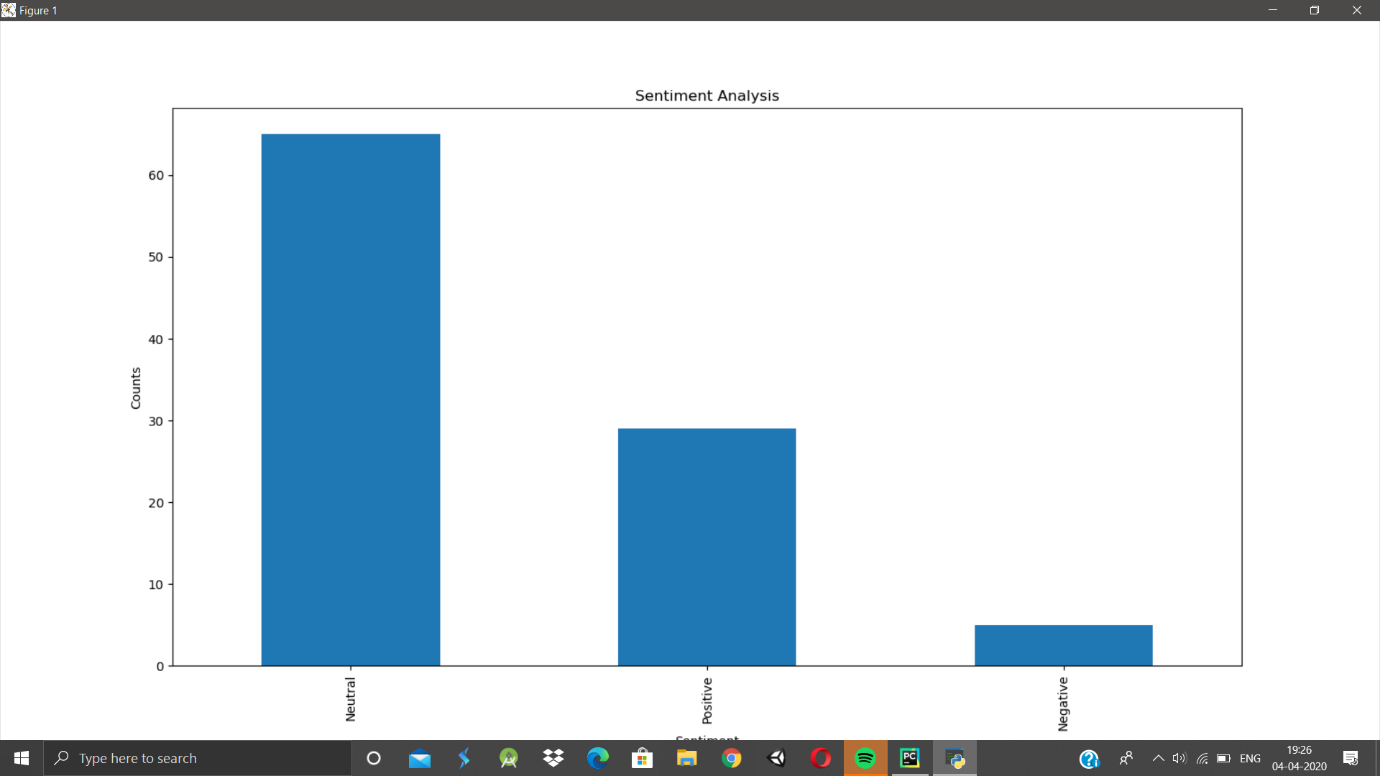


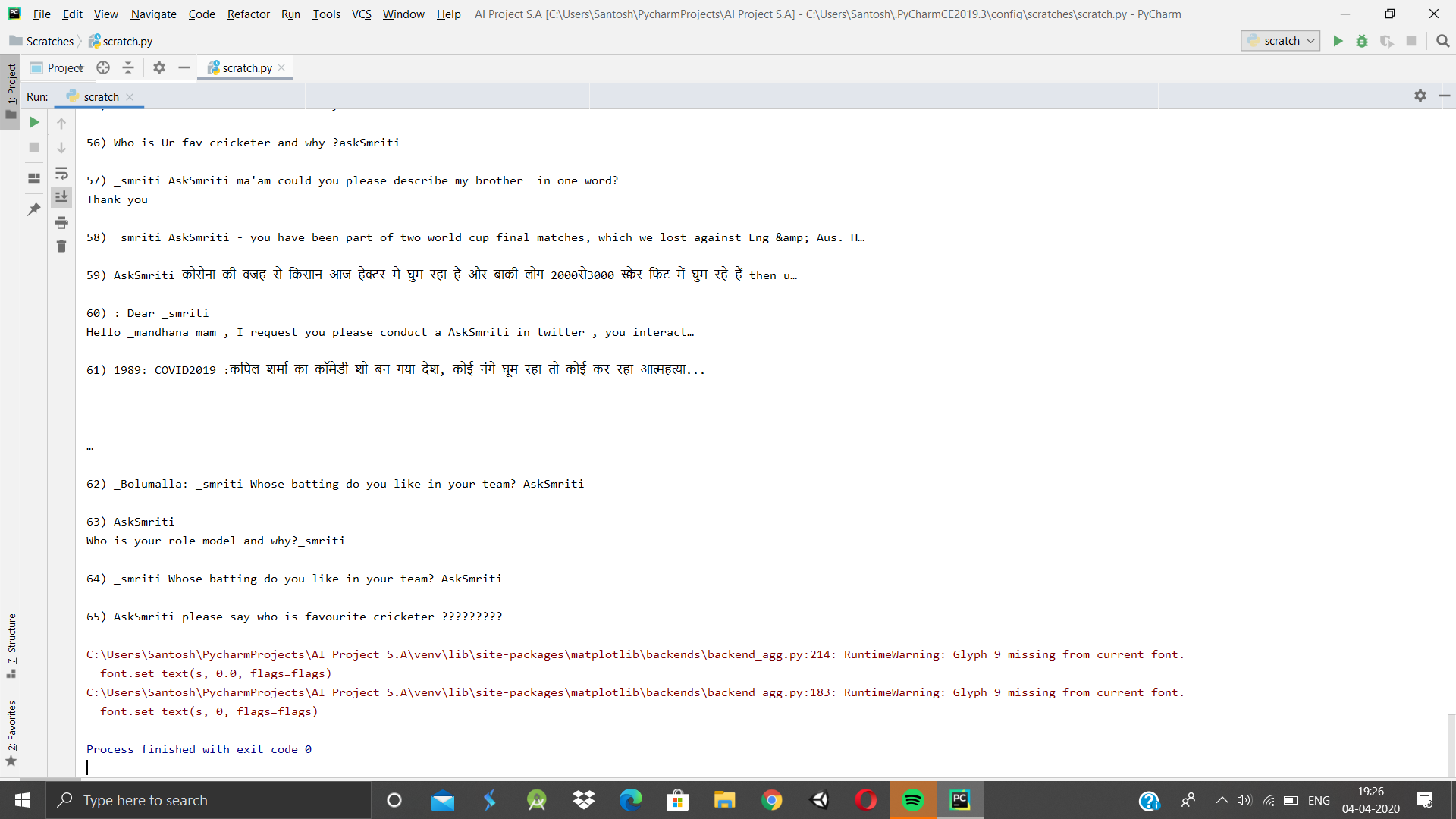
1. Search word : #Asksmriti

Indian women cricketer Smriti Mandhana hosted a questionnaire with #asksmriti hashtag so we will analyse by extracting 100 tweets.



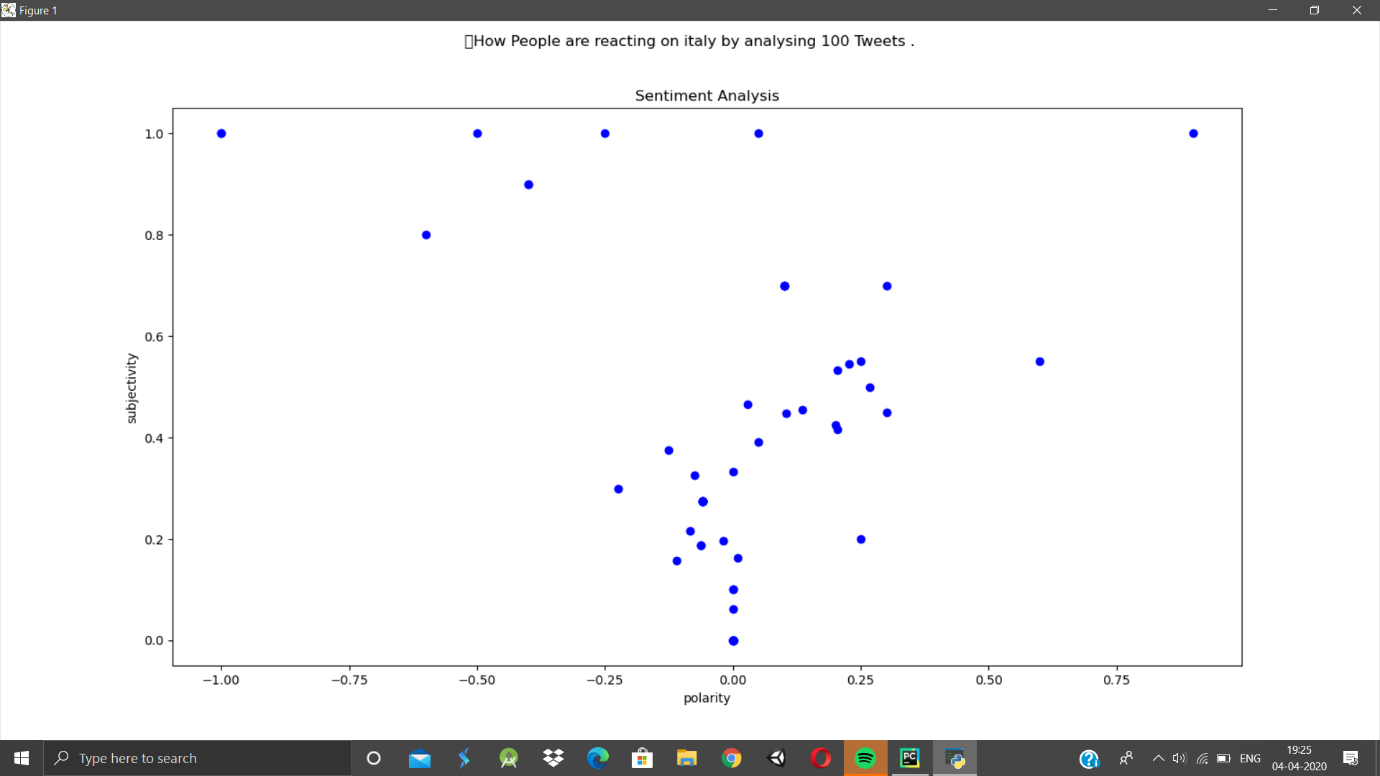
Scatter Plot of Asksmriti tweets



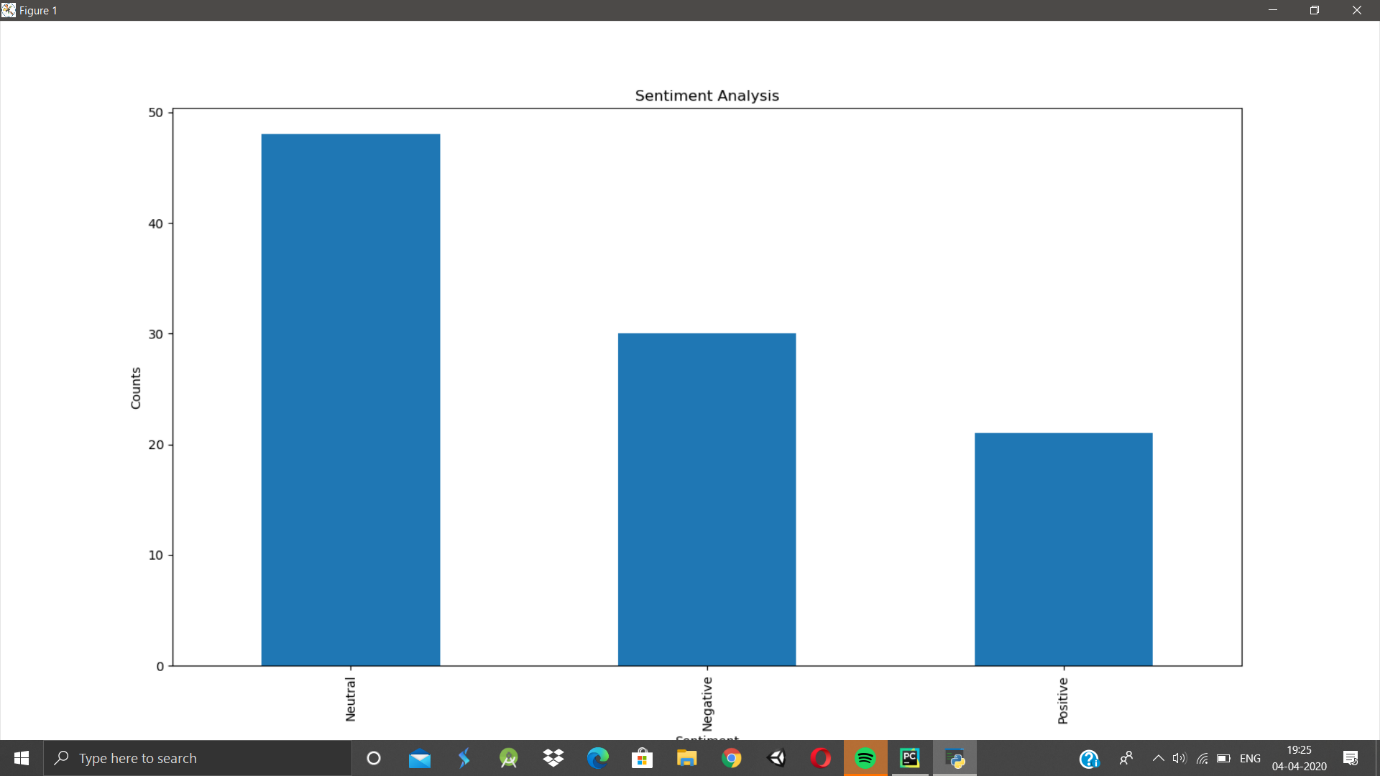


1. Search word :Italy

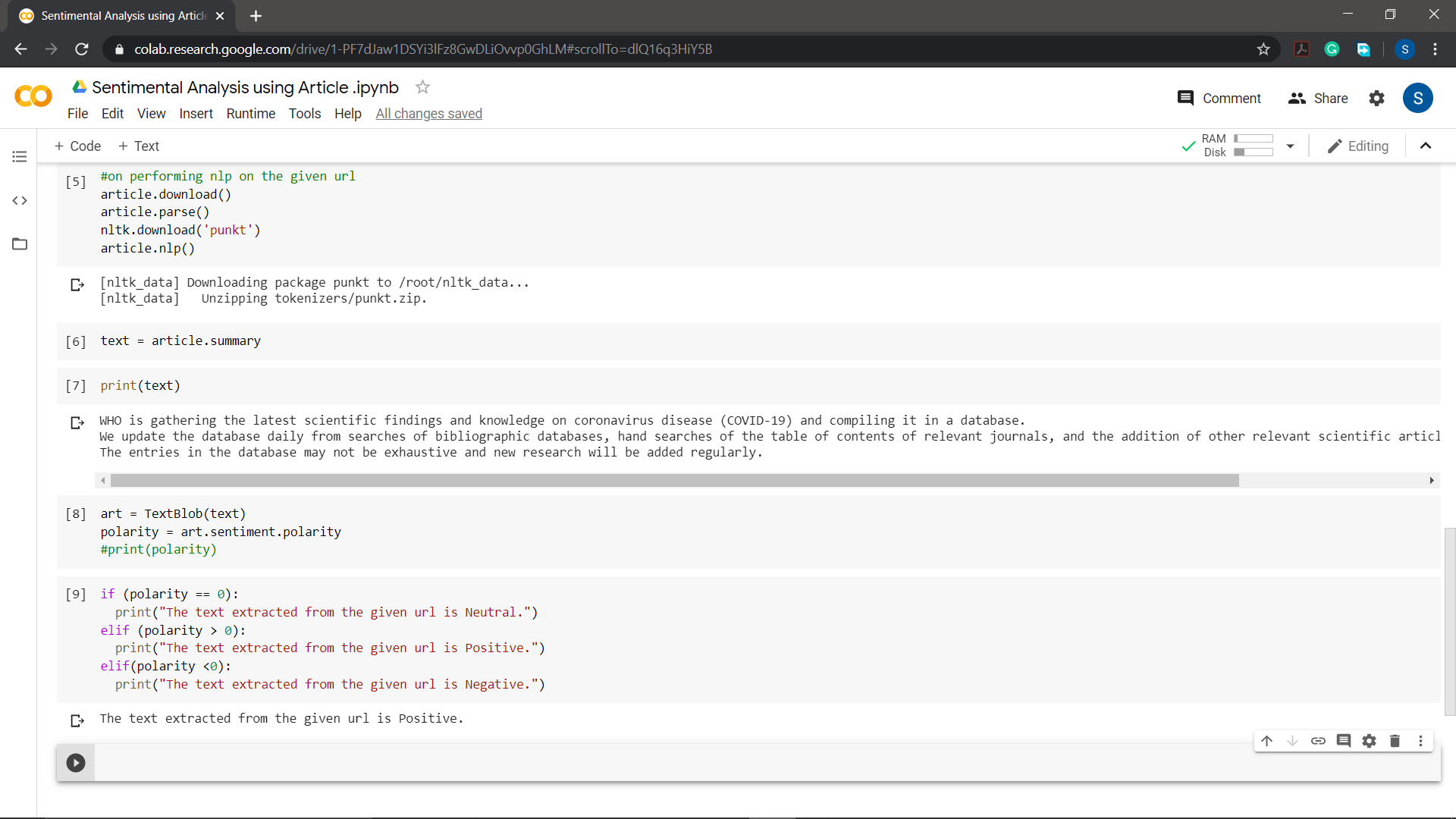
We will analyse tweets with Italy as keyword with 100 tweets.



Scatter Plot of tweets with Italy as keyword.



1. Output of Sentimental Analysis of Online Article.

Finally the output is shown as the text in the article is positive polarity.

**Team Responsibilities:**

**Santosh (25) :** Importing Libraries, twitter api keys and authentication, getting access for api, scatterplot, cleaning of tweets, worked on Pandas (working on DataFrames), sentimental analysis of online article, printing tweets based on polarity.

**Srikanth (02) :** worked on polarity and subjectivity functions , plotted Bar graph,and percentage calculation part of tweets, worked on Pandas.

Libraries:

The back-bone of this project is Natural Language Processing – (NLP)

1. NLTK: - Natural Language Toolkit (NLP).

Natural Language Processing, usually shortened as NLP, is a branch of artificial intelligence that deals with the interaction between computers and humans using the natural language. The ultimate objective of NLP is to read, decipher, nderstand, and make sense of the human languages in a manner that is valuable.

Simply, NLP is a part of Computer Science & Artificial Intelligence which deals with the human languages. NLP refers to the AI method of communicating with Intelligent systems using natural language.

1. tweepy

Tweepy is an API for providing tweets.

Basically It’ll give us four different kind of keys , to access tweets from our account , so that we can access tweets from the twitter and can analyse them for our purpose.

1. textblob

TextBlob is a Python library for processing textual data. It provides a simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more.

1. Pandas

Pandas is a library in python that is used to perform Data Analysis and Data Manipulation.

For example, If we download Datasets from Kaggle , or from any kind of source, so in order to extract the required data from that file we use pandas DataFrame object.

1. numpy

For Processing data in to arrays , and for performing operations on data stored in arrays.

1. re

re means regular expressions.

The ‘re’ package provides multiple methods to perform queries on an input string.

Example:

* re.match()
* re.search()
* re.findall()
* re.sub()
* re.compile()

1. Newspaper3k

This library is used when we are doing sentimental analysis for online articles , it will be able to represent the data precisely ( Summary of the article).

1. Matplot

This library is used to plot the graph based on the polarity and subjectivity of tweets.For plotting the scatter plot and the bargraph for the given set of tweets.

References:

* Edureka YouTube channel.
* <https://www.analyticsvidhya.com/blog/2018/07/hands-on-sentiment-analysis-dataset-python/>
* <https://datahack.analyticsvidhya.com/contest/practice-problem-twitter-sentiment-analysis/>
* <https://www.datacamp.com/community/tutorials/simplifying-sentiment-analysis-python>
* <https://www.youtube.com/watch?v=O_B7XLfx0ic>
* <https://www.youtube.com/watch?v=ujId4ipkBio>
* https://monkeylearn.com/sentiment-analysis/