A Project Report
On

SENTIMENT ANALYSIS ON YOUTUBE DATA

Submitted by:

BIDYUT KONWAR

Under the guidance of

Mr. Prabhat Das
(PGT-IP)

ARMY PUBLIC SCHOOL JORHAT
MES GATE ROWRIAH JORHAT
Assam-785004

CERTIFICATE BY PRINCIPAL

This is to certify that this project report entitled “Sentiment Analysis on Youtube Data” submitted by Bidyut Konwar to Army Public School Jorhat has been examined and evaluated.

His project embodies up to the standards both in respect of its contents and form as per CBSE norms and his original views.

Date:

Place:

Mrs. Ferdausi Sultana Hazarika

(Principal)

Army Public School Jorhat

CERTIFICATE BY EXAMINERS

This is to certify that this project report entitled "**SENTIMENT ANALYSIS ON YOUTUBE DATA**" is the bonafide work of **BIDYUT KONWAR** who carried out the project work under my supervision and guidance.

To the best of my knowledge, the matter embodied in the report has not been submitted to any other institute for the award of any other degree.

Date:
Place:

	Mr. Prabhat Das
(External Examiner)	(Internal Examiner)

ACKNOWLEDGEMENT

I take this opportunity to extend my heart full gratitude to Army Public School Jorhat for providing me the opportunity.

I am highly grateful to my guide Mr. Prabhat Das, PGT-IP, Army Public School Jorhat for giving us the opportunity to work under him and providing us an ample guidance and support through the project.

Lastly, I would also like to thank the authors whose publications guided us regarding our project.

DECLARATION

I admit that this report is of my own work and all the sources of the information used in this report have fully acknowledged.

I hereby declare that the dissertation work entitled “**SENTIMENT ANALYSIS ON YOUTUBE DATA**” submitted to the Army Public School Jorhat, is prepared by me and was not submitted to any other institution for award of any other degree.

Date:
Place:

Signature

Abstract

Sentiment analysis is the computational study of people's opinions, sentiments, attitudes, and emotions expressed in written language. It is one of the most active research areas in natural language processing and text mining in recent years. Its popularity is mainly due to two reasons. First, it has a wide range of applications because opinions are central to almost all human activities and are key influencers of our behaviours. Whenever we need to make a decision, we want to hear other's opinions.

Second, it presents many challenging research problems, which had never been attempted before the year 2000. Part of the reason for the lack of study before was that there was little opinionated text in digital forms. It is thus no surprise that the inception and the rapid growth of the field coincide with those of the social media on the Web.

In fact, the research has also spread outside of computer science to management sciences and social sciences due to its importance to business and society as a whole.

This program is an attempt to analyse the same from a given dataset and also includes a simple yet effective method of registration for users.

TABLE OF CONTENTS

List of Figures

Page no

Tools and Libraries Used

MySQL

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language. A relational database organizes data into one or more data tables in which data types may be related to each other; these relations help structure the data. SQL is a language, programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups.

MySQL is free and open-source software under the terms of the GNU General Public License, and is also available under a variety of proprietary licenses. MySQL was owned and sponsored by the Swedish company MySQL AB, which was bought by Sun Microsystems (now Oracle Corporation). In 2010, when Oracle acquired Sun, Widenius forked the open-source MySQL project to create MariaDB [1].

PyCharm

PyCharm is an integrated development environment (IDE) used in computer programming, specifically for the Python language. It is developed by the Czech company JetBrains. It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control systems (VCSes), and supports web development with Django as well as data science with Anaconda.

PyCharm is cross-platform, with Windows, macOS and Linux versions.

The Community Edition is released under the Apache License, and there is also Professional Edition with extra features – released under a proprietary license [2].

Pandas

Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license. The name is derived from the term "panel data", an econometrics term for data sets that include observations over multiple time periods for the same individuals. Its name is a play on the phrase "Python data analysis" itself. Wes McKinney started building what would become pandas at AQR Capital while he was a researcher there from 2007 to 2010 [3].

Text Blob

TextBlob is a Python (2 and 3) library for processing textual data. It provides a consistent API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, and more [4].

Matplotlib

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK+. There is also a procedural "pylab" interface based on a state machine (like OpenGL), designed to closely resemble that of MATLAB, though its use is discouraged. SciPy makes use of Matplotlib.

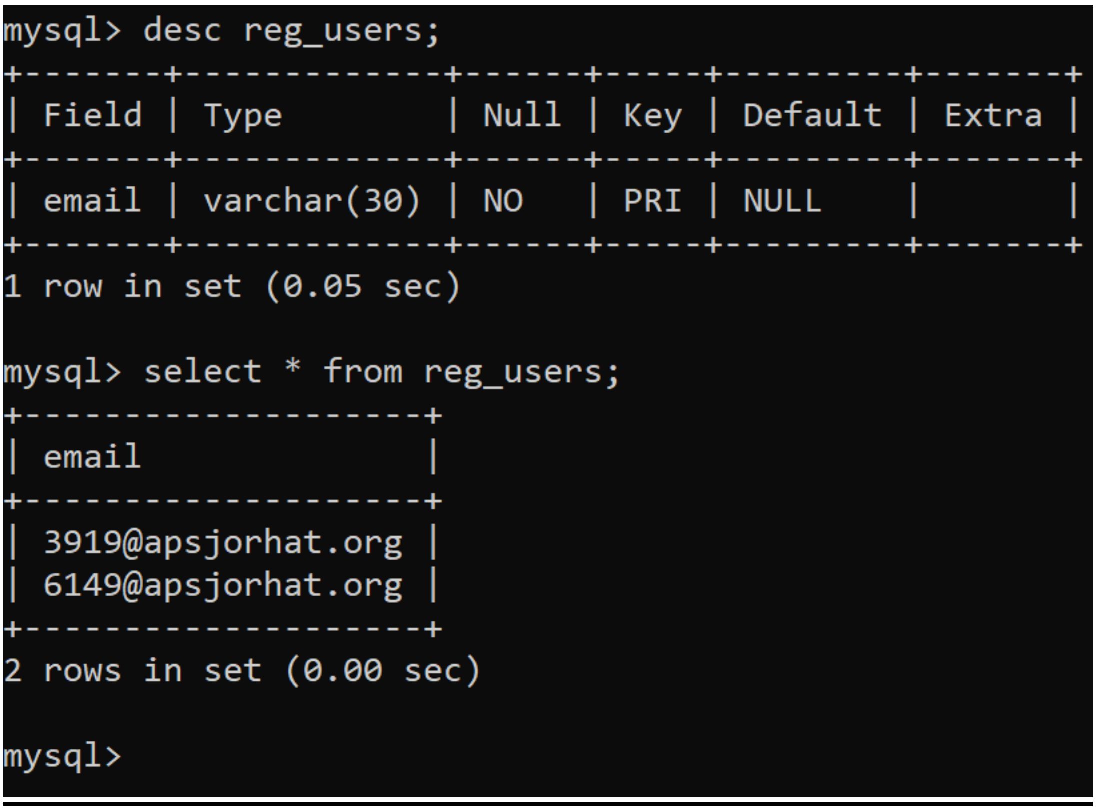
Matplotlib was originally written by John D. Hunter. Since then it has an active development community and is distributed under a BSD-style license [5].

Introduction

Sentiment analysis refers to the use of natural language processing, text analysis and computational linguistics to identify and extract subjective information in source materials. Generally speaking, sentiment analysis aims to determine the attitude of a speaker or a writer with respect to some topic or the overall contextual polarity of a document. The attitude may be his or her judgment or evaluation affective state, or the intended emotional communication. Sentiment analysis is the process of detecting a piece of writing for positive, negative, or neutral feelings bound to it. Humans have the innate ability to determine sentiment; however, this process is time consuming, inconsistent, and costly in a business context It's just not realistic to have people individually read tens of thousands of user customer reviews and score them for sentiment.

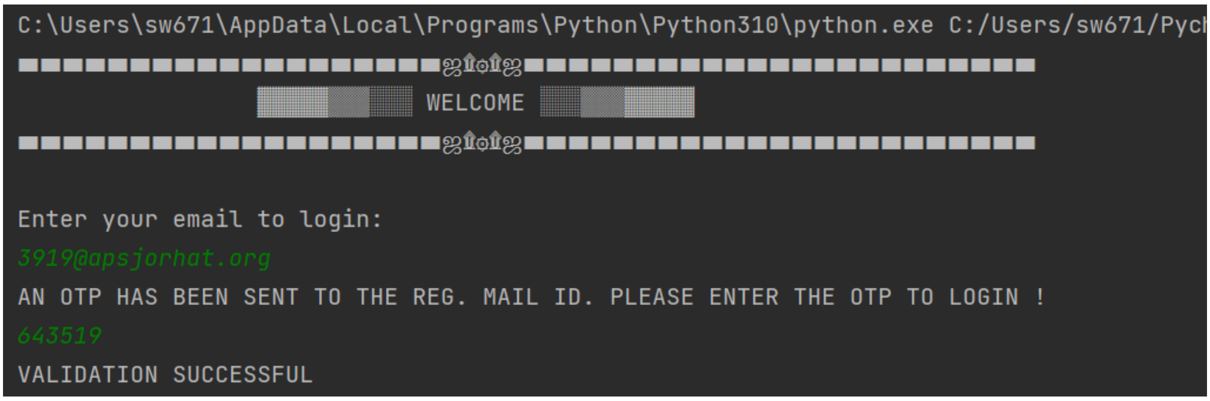
The following program is an attempt to develop a program that can analyse the sentiments expressed by literature from a given dataset and assign its polarity as positive, neutral or negative. It includes a simple registration system where the user can register him/her into the program with the help of an one time password sent to the email address used for the registration process. The program then gives us the number of positive, negative or neutral statements present in the given dataset. The program uses the python library TextBlob for this purpose. A graph is also generated using the Matplotlib library that gives us a visual representation of the polarity of the given data.

Project Overview



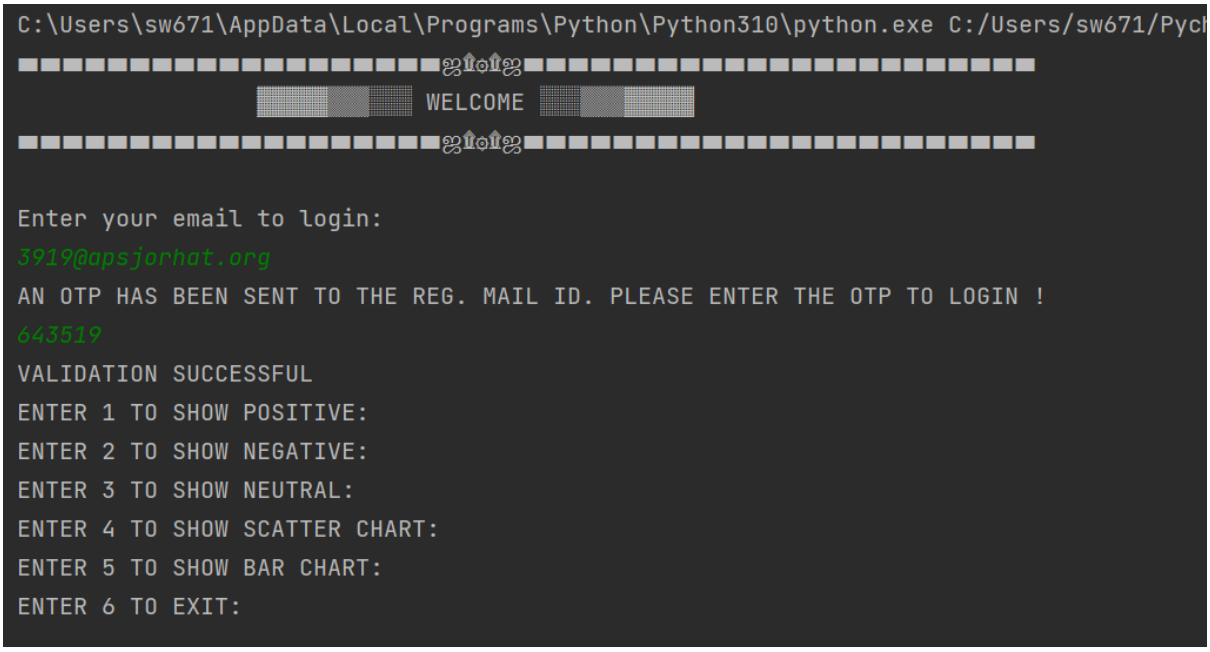
```
mysql> desc reg_users;
+-------+-------------+------+-----+---------+-------+
| Field | Type        | Null | Key | Default | Extra |
+-------+-------------+------+-----+---------+-------+
| email | varchar(30) | NO   | PRI | NULL    |       |
+-------+-------------+------+-----+---------+-------+
1 row in set (0.05 sec)

mysql> select * from reg_users;
+--------------------+
| email              |
+--------------------+
| 3919@apsjorhat.org |
| 6149@apsjorhat.org |
+--------------------+
2 rows in set (0.00 sec)

mysql>
```

Figure 1: All the information regarding the users is stored in this table



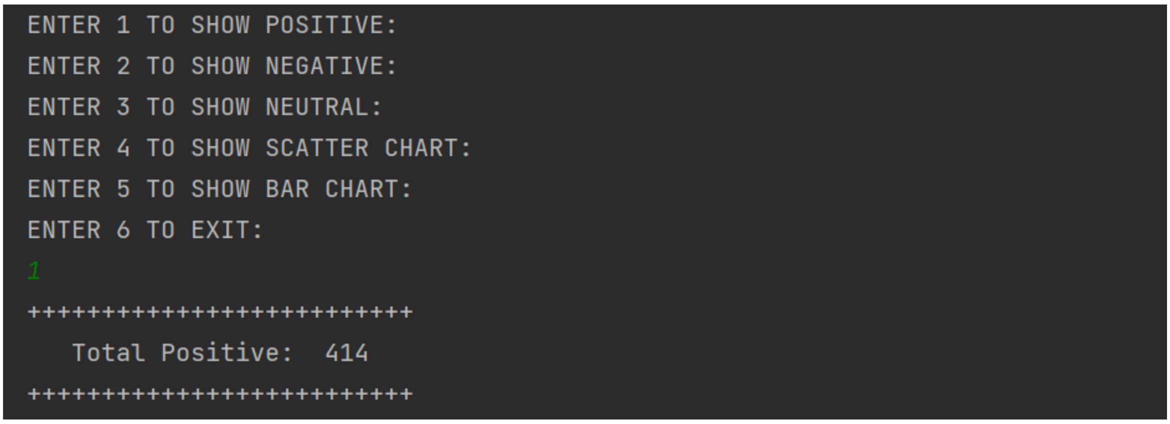
```
C:\Users\sw671\AppData\Local\Programs\Python\Python310\python.exe C:/Users/sw671/Pyc
■■■■■■■■■■■■■■■■■■■ஜ۩۞۩ஜ■■■■■■■■■■■■■■■■■■■■■■■
                 ▒▒▒▓▓░░ WELCOME ░░▓▓▒▒▒
■■■■■■■■■■■■■■■■■■■ஜ۩۞۩ஜ■■■■■■■■■■■■■■■■■■■■■■■

Enter your email to login:
3919@apsjorhat.org
AN OTP HAS BEEN SENT TO THE REG. MAIL ID. PLEASE ENTER THE OTP TO LOGIN !
643519
VALIDATION SUCCESSFUL
```

Figure 2 : When the home page is opened program asks for user’s email address, on entering an email id, program sends an OTP configuration to user’s email.



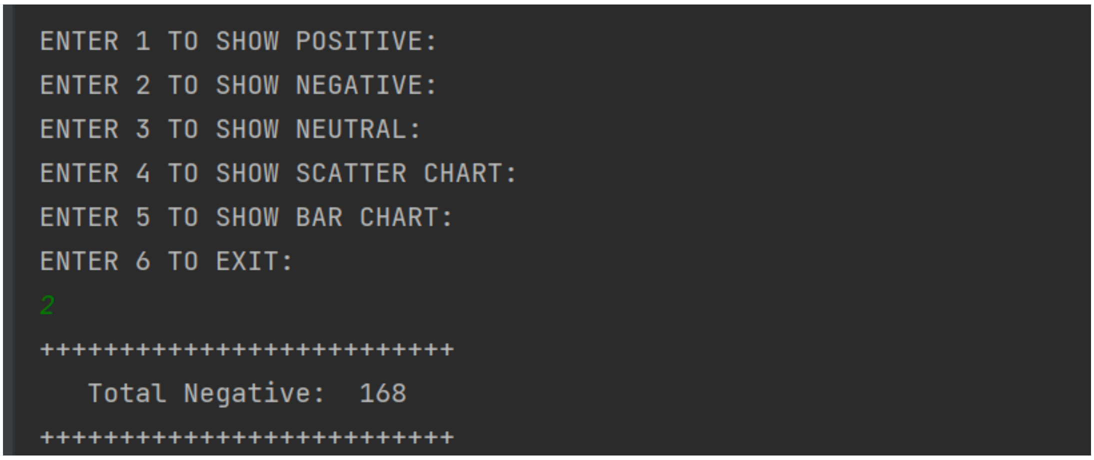
```
C:\Users\sw671\AppData\Local\Programs\Python\Python310\python.exe C:/Users/sw671/Pyc
▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬ஜ۩۞۩ஜ▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬
                 ░░░░▒▒▓▓ WELCOME ▓▓▒▒░░░░
▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬ஜ۩۞۩ஜ▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬

Enter your email to login:
3919@apsjorhat.org
AN OTP HAS BEEN SENT TO THE REG. MAIL ID. PLEASE ENTER THE OTP TO LOGIN !
643519
VALIDATION SUCCESSFUL
ENTER 1 TO SHOW POSITIVE:
ENTER 2 TO SHOW NEGATIVE:
ENTER 3 TO SHOW NEUTRAL:
ENTER 4 TO SHOW SCATTER CHART:
ENTER 5 TO SHOW BAR CHART:
ENTER 6 TO EXIT:
```

Figure 3: After the login procedure is completed the user is presented with the following list of options



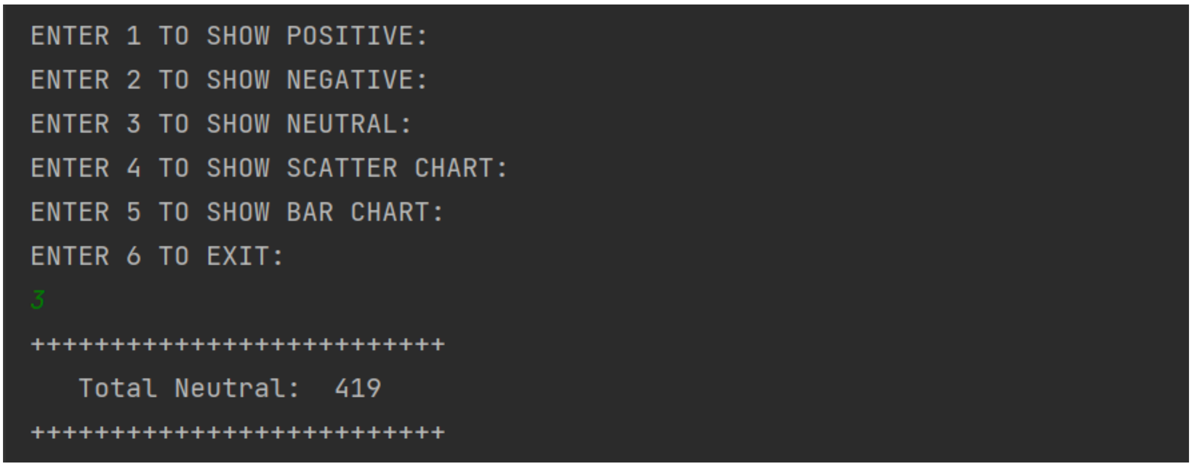
```
ENTER 1 TO SHOW POSITIVE:
ENTER 2 TO SHOW NEGATIVE:
ENTER 3 TO SHOW NEUTRAL:
ENTER 4 TO SHOW SCATTER CHART:
ENTER 5 TO SHOW BAR CHART:
ENTER 6 TO EXIT:
1
++++++++++++++++++++++++++
   Total Positive:  414
++++++++++++++++++++++++++
```

Figure 4: Output for option 1 (Total positive)



```
ENTER 1 TO SHOW POSITIVE:
ENTER 2 TO SHOW NEGATIVE:
ENTER 3 TO SHOW NEUTRAL:
ENTER 4 TO SHOW SCATTER CHART:
ENTER 5 TO SHOW BAR CHART:
ENTER 6 TO EXIT:
2
++++++++++++++++++++++++++
   Total Negative:  168
++++++++++++++++++++++++++
```

Figure 5: Output for option 2 (Total negative)



```
ENTER 1 TO SHOW POSITIVE:
ENTER 2 TO SHOW NEGATIVE:
ENTER 3 TO SHOW NEUTRAL:
ENTER 4 TO SHOW SCATTER CHART:
ENTER 5 TO SHOW BAR CHART:
ENTER 6 TO EXIT:
3
++++++++++++++++++++++++++
   Total Neutral:  419
++++++++++++++++++++++++++
```

Figure 6: Output for option 3 (Total neutral)

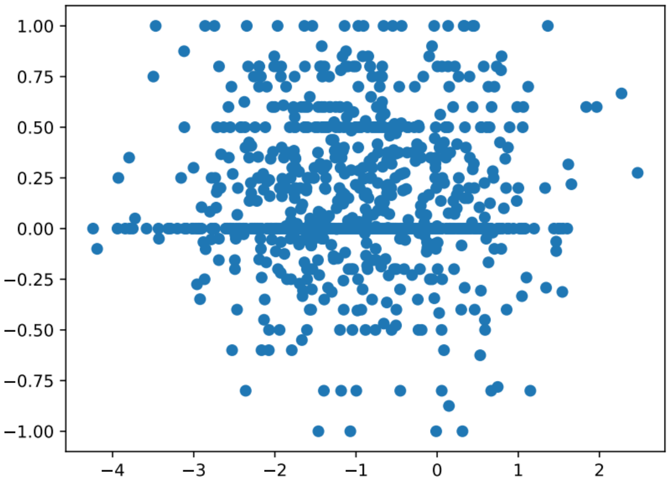


Figure 7: Output for option 4 (Scatter Chart)

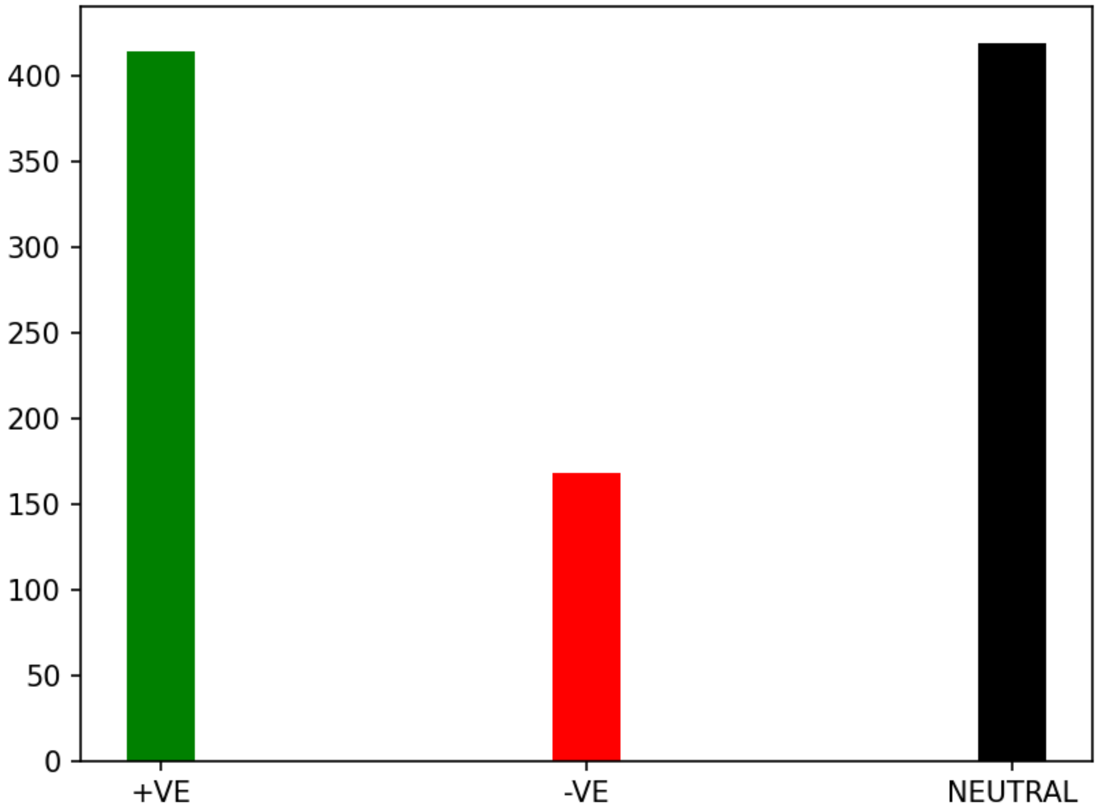
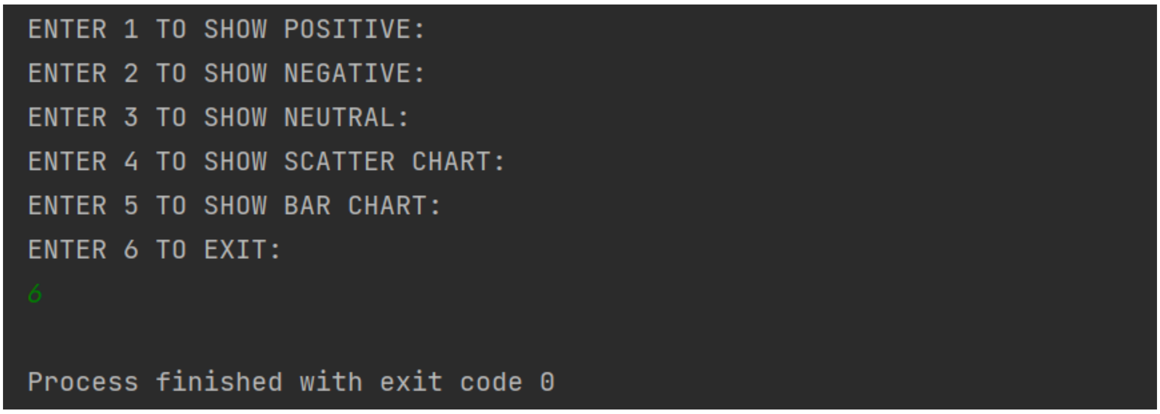


Figure 8: Output for option 5 (Bar Graph)

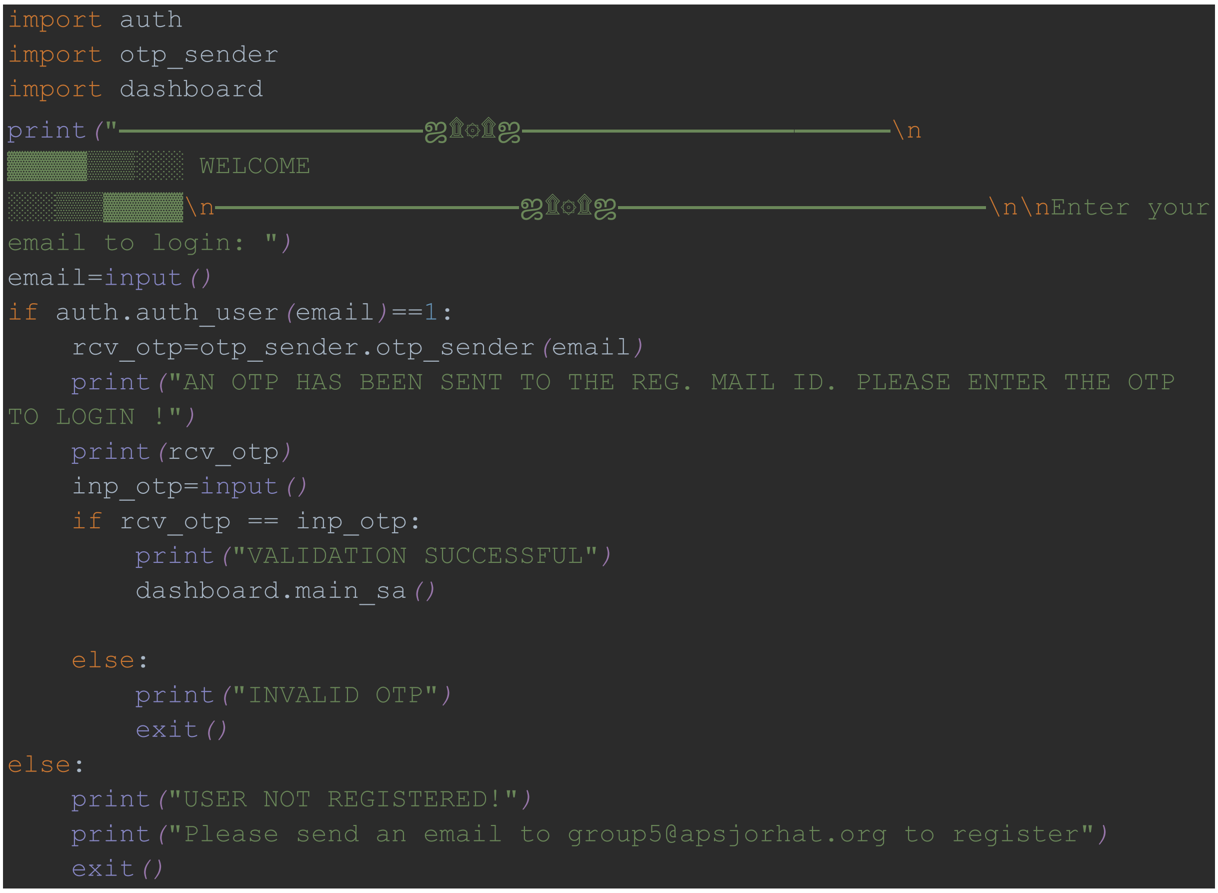


```
ENTER 1 TO SHOW POSITIVE:
ENTER 2 TO SHOW NEGATIVE:
ENTER 3 TO SHOW NEUTRAL:
ENTER 4 TO SHOW SCATTER CHART:
ENTER 5 TO SHOW BAR CHART:
ENTER 6 TO EXIT:
6

Process finished with exit code 0
```

Figure 9: Output for option 6 (Exit)

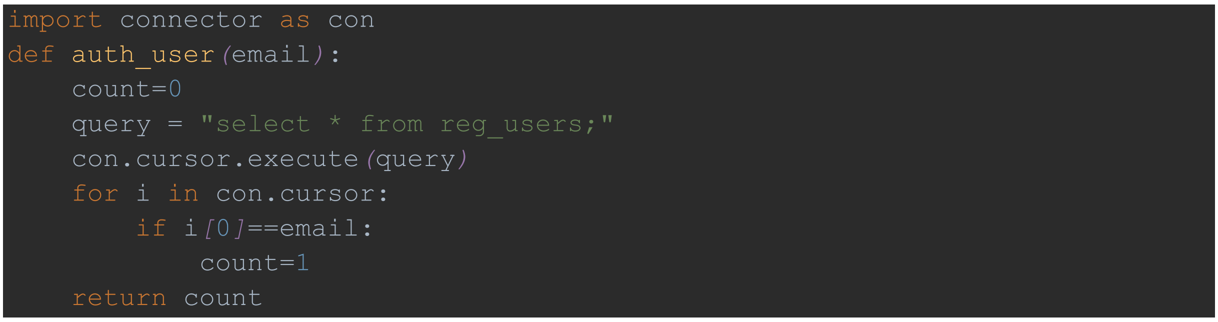
Source Code

main



```
import auth
import otp_sender
import dashboard
print("▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬ஜ۩۞۩ஜ▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬\n
▓▓▓▓▒▒▒░░░ WELCOME
░░░▒▒▒▓▓▓▓\n▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬ஜ۩۞۩ஜ▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬\n\nEnter your
email to login: ")
email=input()
if auth.auth_user(email)==1:
    rcv_otp=otp_sender.otp_sender(email)
    print("AN OTP HAS BEEN SENT TO THE REG. MAIL ID. PLEASE ENTER THE OTP
TO LOGIN !")
    print(rcv_otp)
    inp_otp=input()
    if rcv_otp == inp_otp:
        print("VALIDATION SUCCESSFUL")
        dashboard.main_sa()

    else:
        print("INVALID OTP")
        exit()
else:
    print("USER NOT REGISTERED!")
    print("Please send an email to group5@apsjorhat.org to register")
    exit()
```

Authentication



```
import connector as con
def auth_user(email):
    count=0
    query = "select * from reg_users;"
    con.cursor.execute(query)
    for i in con.cursor:
        if i[0]==email:
            count=1
    return count
```

Connector

```
import mysql.connector as mc
dbc =
mc.connect(host="localhost",user="root",passwd="root",database="yt_data")
cursor=dbc.cursor()
```

OTP Sender

```
import smtplib
import random
def otp_sender(email):
    otp=str(random.randint(100000,999999))
    SUBJECT = 'OTP FOR LOGIN'
    TEXT = 'YOUR OTP TO LOGIN IS:' + otp
    s = smtplib.SMTP('smtp.gmail.com', 587)
    s.starttls()
    s.login('group5@apsjorhat.org', 'apsj#12345678')
    message = 'Subject:{} \n\n{}'.format(SUBJECT, TEXT)
    s.sendmail('group5@apsjorhat.org', email, message)
    s.quit()
    return otp
```

Dashboard:

```
import textblob as tb
import matplotlib.pyplot as plt
import numpy as np
import csv
def main_sa():
    print("ENTER 1 TO SHOW POSITIVE:\r\nENTER 2 TO SHOW NEGATIVE:\r\nENTER
3 TO SHOW NEUTRAL:\r\nENTER 4 TO SHOW SCATTER CHART:\r\nENTER 5 TO SHOW BAR
CHART:\r\nENTER 6 TO EXIT:")
    delimiters = ["[", "'", "]", "(", ")"]
    pos = 0
    neg = 0
    neu = 0
    y = []
    a = int(input())
    with open('yt_comments.csv', 'r',errors='ignore') as file:
        reader = csv.reader(file)
        for row in reader:
            data = row
            string_data = str(data)

            for i in delimiters:
                string_data = string_data.replace(i, '')
            input_to_textblob = tb.TextBlob(string_data)
            sentence_polarity = input_to_textblob.sentiment.polarity

            if (sentence_polarity > 0):
                y.append(sentence_polarity)
                pos += 1
            elif (sentence_polarity == 0):
                y.append(sentence_polarity)
                neu += 1
            elif (sentence_polarity < 0):
                y.append(sentence_polarity)
                neg += 1

    if a == 1:
        print("++++++++++++++++++++++++++\n   Total Positive: 
",pos,"\n++++++++++++++++++++++++++")
        main_sa()
    elif a == 2:
        print("++++++++++++++++++++++++++\n   Total Negative: ", 
neg,"\n++++++++++++++++++++++++++")
        main_sa()
    elif a == 3:
        print("++++++++++++++++++++++++++\n   Total Neutral: ", neu, 
"\n++++++++++++++++++++++++++")
        main_sa()
    elif a == 4:
        x = np.random.normal(min(y), max(y), len(y))
        plt.scatter(x, y)
        plt.savefig("scatter_sentiment_analysis.pdf")
        plt.show()
        main_sa()
    elif a == 5:
       x=[5,10,15]
       y=[pos,neg,neu]
       plt.bar(x,y,color=['g','r','k'])
       plt.xticks(x,['+VE','-VE','NEUTRAL'])
       plt.savefig("bar_sentiment_analysis.pdf")
       plt.show()
       main_sa()
    elif a==6:
        exit()
```

COMMANDS USED IN MySQL

Creating database

```
Create database yt_data;
```

Using database

```
Use yt_data;
```

Creating table and inserting values

```
Create table reg_users(email varchar(30));
```

Desc table

```
Desc reg_users;
```

Inserting values

```
insert into reg_users values(("3919@apsjorhat.org"),(“6149@apsjorhat.org")) ;
```

To fetch all values

```
Select * from reg_users;
```

Conclusion and Future Work

In this project we have created a program that can be used to analyse the sentiments expressed by literature. The analysed statements are then assigned a polarity that describes whether the statement is positive, negative or neutral.

Most of the world’s data is unstructured and unorganised and a program such as ours can help sort the data in an efficient manner and generate useful information in the form of the sentiments expressed by people.

Sentiment analysis is extremely important because it helps businesses quickly understand the overall opinions of their customers. By automatically sorting the sentiment behind reviews, social media conversations, and more, we can make faster and more accurate decisions.

The field of sentiment analysis is an exciting new research direction due to large number of real-world applications where discovering people’s opinion is important in better decision-making. What we have developed here is an elementary sentiment analysis program. There is a lot of scope for upgrading our program, we can use advanced sentiment analysis models that are more accurate with better semantic knowledge. We can also add the functionality to link our program with social media sites for sentiment analysis of social media posts. There is also the scope to develop a web-based application for sentiment analysis.

References

[1] Wikipedia, https://en.wikipedia.org/wiki/MySQL, Accessed on 20-01-2022

[2] Wikipedia, https://en.wikipedia.org/wiki/PyCharm, Accessed on 20-01-2022

[3] Wikipedia, https://en.wikipedia.org/wiki/Pandas_(software), Accessed on 20-01-2022

[4] Text Bob, https://textblob.readthedocs.io/en/dev/, Accessed on 20-01-2022

[5] Wikipedia, https://en.wikipedia.org/wiki/Matplotlib, Accessed on 20-01-2022