Acknowledgement

It is matter of great pleasure to present this project where positive efforts are translated into some concrete things. We went through many problems and challenges during our work. Many of the challenges that we faced have helped us to learn new things. This project would not have been possible without the guidance and support - technical and personal - of a number of people.

Our sincere thanks to respected Prof. Ashish Patel for modeling our thoughts and vision towards this project and which helped us a great deal during the project. We have referenced some books and many articles on the web which proved to be immensely helpful to us. Finally a special thanks to our friends and classmates for the support, advice and encouragement throughout the tenure of this project.

Arpan Nariya

Briyan Hingarajia

Pratip Patel

Priyank Patel

Abstract

Chat analysis deals with the process of analyzing text to gather information which is useful. In today's developing world, a text has became the most common means of communication. Chat analysis generally refers to analyze large natural language text and which detects usage patterns to extract useful information.

In this project we analyze WhatsApp chat messages and plot a pie chart visualizing the percentage of texts that have a Positive, Negative and Neutral connotation or Sentiment to it. We will also learn how to analyze WhatsApp chats using Python...

Text Mining is just a fancy term for deriving super-awesome patterns and drawing amazing inferences from Textual Data. Just like you can look at an image and infer that it is of a baby or of a 77-year-old woman, we can do the same with texts.

TABLE OF CONTENTS

List of Figures

Chapter 1: Introduction

1.1 Introduction

The world is moving towards a fully digitalized economy at an incredible pace and as a result, a ginormous amount of data is being produced by the internet, social media, smartphones, tech equipment and many other sources each day which has led to the evolution of Big Data management and analytics. Sentiment analysis is one such tool and the most popular branch of textual analytics which with the help of statistics and natural language processing examine and classify the unorganized textual data into various sentiments. It is also known as opinion mining as it largely focuses on the opinion and attitude of the people through analyzing their texts.

More than 34 billion texts are exchanged over the WhatsApp every day and just imagine if we could analyze and get valuable insights from this data and leverage it to not only take better real-time decisions but also add value to the stakeholders at much lower cost and time and hence align our operational efficiency with organizational strategy. In this article, we'll leverage the power of sentiment analysis to investigate the WhatsApp chat using R, visualize and interpret the results at the same time.

WhatsApp is most popular chat app with monthly active users of more than 700 million. The popularity of this app has made it a necessary app among smartphone users and even businesses and organizations use WhatsApp for daily communication in groups and across departments. Corporations get a huge amount of textual data from WhatsApp and they can leverage WhatsApp chat sentiment analysis to gain better insights about their employees and try to avoid unforeseen conflicts due to various redundancies and inefficiency of business processes.

Following figure give us a brief about how whole process execute.

First of all user need to export his/her chat in text file than upload that text file in to the code after uploading file, system will analyze the chat and give output in pie chart. This is short explanation of system.

1.2 Scope

We will use metadata which we deleted and draw awesome inferences from that to study and analyze Whatsapp chat messages using Python.

We are keen to deliver accurate analysis of the chat by using technology such as python, web and other. If you have ever emailed a WhatsApp chat to yourself, (or someone else, if that's how you roll) you may have noticed that it includes a handful of details that can be used to analyze a group of texts as well as the entire chat history as .txt file. Like any other project, there are a few dependencies (libraries) that you would need to have in order to analyze WhatsApp chats using Python.

Sentiment analysis largely depends on the predefined words and their individual score. Which leads to many problems like ambiguity in the context of the sentence? A sentence which includes 'good' might not have any emotions attached to it but will be shown as positive by the analysis.

1.3 Project Summary, Purpose & Overview

In future we are going to build one UI for this project; it gives user a pie chart of analyzed data of exported WhatsApp chat file.

We find the field—"neg", "pos" or "neu"—which has the maximum score and increment the counter keeping track of it.

1.4 Problem Definition

WhatsApp chat analysis using python.

Chapter 2: Technology & Literature Review

2.1 About Tools & Technology

2.1.1 NLTK Sentiment Analysis — About NLTK:

This project is mainly depend on python 3 and natural language toolkit (nltk) module, lets discuss what is NLTK.

- 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.
- It was developed by Steven Bird and Edward Loper in the Department of Computer and Information Science at the University of Pennsylvania.

NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum.

Thanks to a hands-on guide introducing programming fundamentals alongside topics in computational linguistics, plus comprehensive API documentation, NLTK is suitable for linguists, engineers, students, educators, researchers, and industry users alike. NLTK is available for Windows, Mac OS X, and Linux. Best of all, NLTK is a free, open source, community-driven project.

NLTK has been called “a wonderful tool for teaching, and working in, computational linguistics using Python,” and “an amazing library to play with natural language.”

Natural Language Processing with Python provides a practical introduction to programming for language processing. Written by the creators of NLTK, it guides the reader through the fundamentals of writing Python programs, working with corpora, categorizing text, analysing linguistic structure and more. The online version of the book has been updated for Python 3 and NLTK 3.

Sentiment Analysis

Sentiment Analysis is a branch of computer science and overlaps heavily with Machine Learning, and Computational Linguistics Sentiment Analysis is the most common text classification tool that analyses an incoming message and tells whether the underlying sentiment is positive, negative, or neutral.

It is the process of computationally identifying and categorizing opinions expressed in a piece of text, especially in order to determine whether the writer's attitude towards a particular topic, product, etc. is positive, negative, or neutral.

Sentiment Analysis is a concept of Natural Language Processing and sometimes referred to as opinion mining, although the emphasis, in this case, is on extraction.

Fig:2.1 Sentiment Process

Examples of the sentimental analysis are as follows:

- Is this product review positive or negative?
- Is this customer email satisfied or dissatisfied?
- Based on a sample of tweets, how are people responding to this ad campaign/product release/news item?
- How have bloggers' attitudes about the president changed since the election?
- The purpose of this sentiment analysis is to automatically classify a tweet as a positive or negative Tweet Sentiment wise
- Given a movie review or a tweet, it can be automatically classified in categories. These categories can be user-defined (positive, negative) or whichever classes you want.
- Sentiment Analysis for Brand Monitoring
- Sentiment Analysis for Customer Service
- Sentiment Analysis for Market Research and Analysis

Sample Positive Tweets

- I love this car
- This view is amazing
- I feel great this morning
- I am so excited about the concert
- He is my best friend

Sample Negative Tweets

- I do not like this car
- This view is horrible
- I feel tired this morning
- I am not looking forward to the concert
- He is my enemy

Sentimental Analysis Process

- The list of word features needs to be extracted from the tweets
- It is a list with every distinct word ordered by frequency of appearance
- The use of Feature Extractor to decide which features are more relevant
- The one we are going to use returns a dictionary indicating that words are contained in the input passed

2.1.2 Matplotlib

Humans are very visual creatures: we understand things better when we see things visualized. However, the step to presenting analyses, results or insights can be a bottleneck: you might not even know where to start or you might have already a right format in mind, but then questions like "Is this the right way to visualize the insights that I want to bring to my audience?" will have definitely come across your mind.
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+.
Typically, a **Pie** Chart is used to show parts to the whole, and often a % share. Luckily for us, **Matplotlib** handles the sizes of the slices and everything, we just feed it the numbers. Within the plt.**pie**, we specify the "slices," which are the relevant sizes for each part.

When you're working with the Python plotting library Matplotlib, the first step to answering the above questions is by building up knowledge on topics like:

- The anatomy of a Matplotlib plot: what is a subplot? What are the Axes? What exactly is a figure?

- Plot creation, which could raise questions about what module you exactly need to import (pylab or pyplot?), how you exactly should go about initializing the figure and the Axes of your plot, how to use matplotlib in Jupyter notebooks, etc.

- Plotting routines, from simple ways to plot your data to more advanced ways of visualizing your data.

- Basic plot customizations, with a focus on plot legends and text, titles, axes labels and plot layout.
- Saving, showing, clearing, … your plots: show the plot, save one or more figures to, for example, pdf files, clear the axes, clear the figure or close the plot, etc.
- Lastly, you'll briefly cover two ways in which you can customize Matplotlib: with style sheets and the rc settings.

(To practice matplotlib interactively, try the free Matplotlib chapter at the start of this Intermediate Python course or see DataCamp's Viewing 3D Volumetric Data With Matplotlib tutorial to learn how to work with matplotlib's event handler API.)

What Does A Matplotlib Python Plot Look Like?

At first sight, it will seem that there are quite some components to consider when you start plotting with this Python data visualization library. You'll probably agree with me that it's confusing and sometimes even discouraging seeing the amount of code that is necessary for some plots, not knowing where to start yourself and which components you should use.

Luckily, this library is very flexible and has a lot of handy, built-in defaults that will help you out tremendously. As such, you don't need much to get started: you need to make the necessary imports, prepare some data, and you can start plotting with the help of the `plot()` function! When you're ready, don't forget to show your plot using the `show()` function.

Chapter 3: System Requirements Study

3.1 Hardware & Software Requirements

This project has minimal system requirements.

Hardware Requirements: (approx.)

1. Intel Pentium or above
2. 450 MHz Processor
3. 1 GB Ram
4. Minimum Free Hard Drive Space for nltk:500mb
5. Super VGA monitors(800*600)

Software Requirements:

1. Windows XP or higher.
2. Preferably python3 or higher.
3. Preferably nltk and matplotlib version

3.2 File Formats

Input file Formats:

Supported File Formats: .txt file

Supported Languages : English (local language is not supported in current implementation.)

Output Formats: Pie chart, Percentage of word's sentiments

3.3Constraints

3.3.1 Regulatory Policies

- There were no technical regulatory policies specifically covering the application.

3.3.2 Hardware Limitations

- The major limitation would be the number of client request, which our workstation could handle at a time. Also, a major challenge could be management of such a huge database.

3.3.3 Higher Order Language Requirements

- There are no specific high-level language requirements, other than that a high-level language be used to develop and maintain application for more cost-effective maintenance efforts.

3.3.4 Reliability Requirements

- System shall be available for maximum time.
- System shall be robust enough to have a high degree of fault tolerance. The system should not crash in case of high CPU usage.

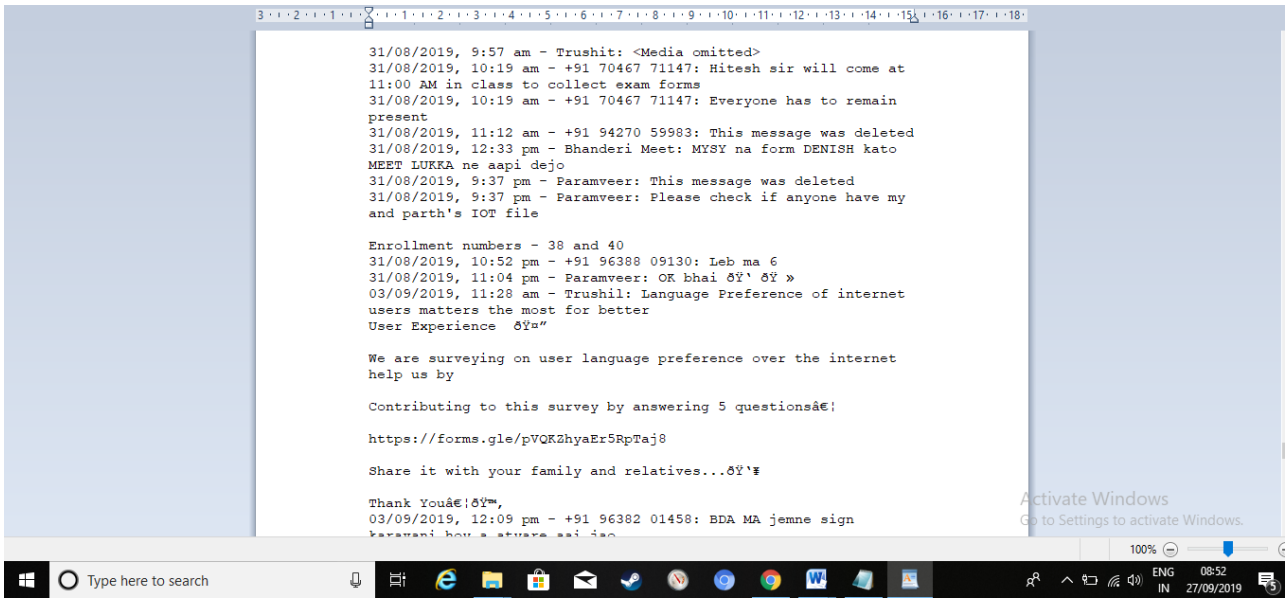
3.4 DATA SET:-

In WhatsApp, one can download chats manually. To export a copy of the history of an individual chat or group, use the Export chat feature:
1. Open the chat for the individual or group.
2. Tap More options
3. Tap More.
4. Tap Export Chat.
5. Choose whether to Include Media or not.
An email will be composed with your chat history attached as a .txt document
Note:
· If you choose to attach media, the most recent media sent will be added as attachments.
· When sending with media, you can send up to 10,000 latest messages. Without media, you can send 40,000 messages. These constraints are due to maximum email sizes.

Format :

DATE, TIME – Sender's name: Massage
For e.g. 06/02/2019, 11:40 pm - Arpan: abcdefghij

In massages, one can send text, numbers, videos, images, recording, symbols and emoji. People can save chat as .txt format and use it as a backup file as well as for analysis purposes like us. From that file one can get the date and time of the chat and who is the author of that particular text.



31/08/2019, 9:57 am - Trushit: <Media omitted>
31/08/2019, 10:19 am - +91 70467 71147: Hitesh sir will come at 11:00 AM in class to collect exam forms
31/08/2019, 10:19 am - +91 70467 71147: Everyone has to remain present
31/08/2019, 11:12 am - +91 94270 59983: This message was deleted
31/08/2019, 12:33 pm - Bhanderi Meet: MYSY na form DENISH kato MEET LUKKA ne aapi dejo
31/08/2019, 9:37 pm - Paramveer: This message was deleted
31/08/2019, 9:37 pm - Paramveer: Please check if anyone have my and parth's IOT file

Enrollment numbers - 38 and 40
31/08/2019, 10:52 pm - +91 96388 09130: Leb ma 6
31/08/2019, 11:04 pm - Paramveer: OK bhai ðŸ‘ ðŸ »
03/09/2019, 11:28 am - Trushil: Language Preference of internet users matters the most for better
User Experience ðŸ¤”

We are surveying on user language preference over the internet help us by

Contributing to this survey by answering 5 questionsâ€¦

https://forms.gle/pVQKZhyaEr5RpTaj8

Share it with your family and relatives...ðŸ‘¥

Thank Youâ€¦ðŸ™,
03/09/2019, 12:09 pm - +91 96382 01458: BDA MA jemne sign karavani hoy a atyare aapi jao

Activate Windows
Go to Settings to activate Windows.

100%

Type here to search

ENG
IN
08:52
27/09/2019

Chapter 4: System Analysis

(Here we are not including class diagram and ER diagram because we don't have any data base here and we are implementing only one feature of WhatsApp.)

The next phase in the System Development life cycle is a system design. The designing part begins after the analysis of the system and is aimed at defining how to do the things. Any design has to be constantly evaluated to ensure that it meets the requirements, is practical and workable in the given environment. If there are a number of alternatives, then all alternatives are evaluated and the best possible solution is implemented.

- **Approaches to Design:-**

There are two main approaches to design, which are:

1) Data Centered Approach.
2) Process Centered Approach.

In both the approaches, the other factor cannot be ignored i.e. process cannot be ignored in data centered approach and vice versa. The data centered approach starts from data structures first and then the processes and the process centric approach aim at defining all the processes first and data structure at the end. Both the approaches have their advantages and disadvantages.

We use the Data Centered approach in the design of the system. The DataFlow Diagram and the Entity-Relationship diagram form the basic input to the design phase. The Data Centered approach is the principal of Object Oriented Design where a collection of data elements and its associated characteristics (processes) are defined as objects.

4.1 Requirement Analysis

The application requirement analysis is produced at the culmination of the analysis task. The function and performance allocated to software as part of system engineering are refined by establishing a complete information description, a detailed functional and behavioural description, an indication of performance requirements and design constraints, appropriate validation criteria, and other data pertinent to requirements.

4.2 Feasibility Study

Feasibility is a practical extent to which a project can be performed successfully. To evaluate feasibility, a feasibility study is performed, which determines whether the solution considered to accomplish the requirements is practical and workable in the software or not. Such information as resource availability, cost estimate for software development, benefits of the software to organization, and cost to be incurred on its maintenance are considered. The objective of the feasibility study is to establish the reasons for developing software that is acceptable to users and adaptable to change.

4.2.1 Operational Feasibility Study

Using this intranet-based application one can connect from anywhere at any time.

The maps are available at any time.

• Some operational features of the city on the road map:

1. Performance
2. Information
3. Economy

Performance: The application provides the connectivity to all the assets in a minimum time required.

Information: This application will provide the detailed information about different assets available with the company.

Economy: This System will use some existence resources like Computers, software, etc.

Cost of this project is depending upon the man hour dedicated to the proposed System.

• Our system provides adequate and its response time is very quick.

• You can easily fetch the data from database.

• The system provides administrator with timely, accurate and usefully formatted information.

4.2.2 Technical Feasibility

This is concerned with specifying equipment and software that will successfully satisfy the user requirement. The technical needs of the system may vary considerably, but might include: • The facility to produce outputs in a given time.

• Response time under certain conditions.

• Ability to process a certain volume of transaction at a particular speed.

• Facility to communicate data to distant locations.

In examining technical feasibility, configuration of the system is given more importance than the actual make of hardware. The configuration should give the complete picture about the system's requirements. How many workstations are required, how these units are interconnected so that they could operate and communicate smoothly?

4.2.3 Scheduling Feasibility

A project will fail if it takes too long to be completed before it is useful. Typically, this means estimating how long the system will take to develop, and if it can be completed in a given time period using some methods like payback period. Schedule feasibility is a measure of how reasonable the project timetable is. Given our technical expertise, are the project deadlines reasonable? Some projects are initiated with specific deadlines. You need to determine whether the deadlines are mandatory or desirable.

4.2.4 Economical Feasibility

Economic analysis could also be referred to as cost/benefit analysis. It is the most frequently used method for evaluating the effectiveness of a new system. In economic analysis the procedure is to determine the benefits and savings that are expected from a candidate system and compare them with costs. If benefits outweigh costs, then the decision is made to design and implement the system. An entrepreneur must accurately weigh the cost versus benefits before taking an action.

The concerned business must be able to see the value of the investment it is pondering before committing to an entire system study. If short-term costs are not overshadowed by long-term gains or produce no immediate reduction in operating costs, then the system is not economically feasible, and the project should not proceed any further. If the expected benefits equal or exceed costs, the system can be judged to be economically feasible. Economic analysis is used for evaluating the effectiveness of the proposed system.

The economic feasibility will review the expected costs to see if they are in-line with the projected budget or if the project has an acceptable return on investment. At this point, the projected costs will only be a rough estimate. The exact costs are not required to determine economic feasibility. It is only required to determine if it is feasible that the project costs will fall within the target budget or return on investment. A rough estimate of the project schedule is required to determine if it would be feasible to complete the systems project within a required timeframe. The required timeframe would need to be set by the organization.

4.2.5 Implementation Feasibility

The main purpose of checking Operational Feasibility is to find out whether the system will be functional after its development and installation or not. The outcomes of the operational feasibility are as follows:

Service Provider website helps the user for getting information about new available service product quickly by presenting the data in proper format online. Graphical representation of data helps the user in taking proper decision in time. So, it is supposed to improve the working efficiency of user.

4.3 Study of Current System

4.3.1 Algorithm

i. Read the file as a “chat_sam”.

ii. Recognize media pattern and date pattern.

iii. Remove media and date from file.

iv. Split the remaining data by lines and store it in list.

v. Initialize neutral, negative and positive as 0, 0, 0.

vi. Find the priority of all the words by sentiment analyzer (it is class of nltk module).

vii. If the priority_score (function of sentiment analyzer) give positive value than increase positive by 1.

viii. If value is negative increase negative by 1.

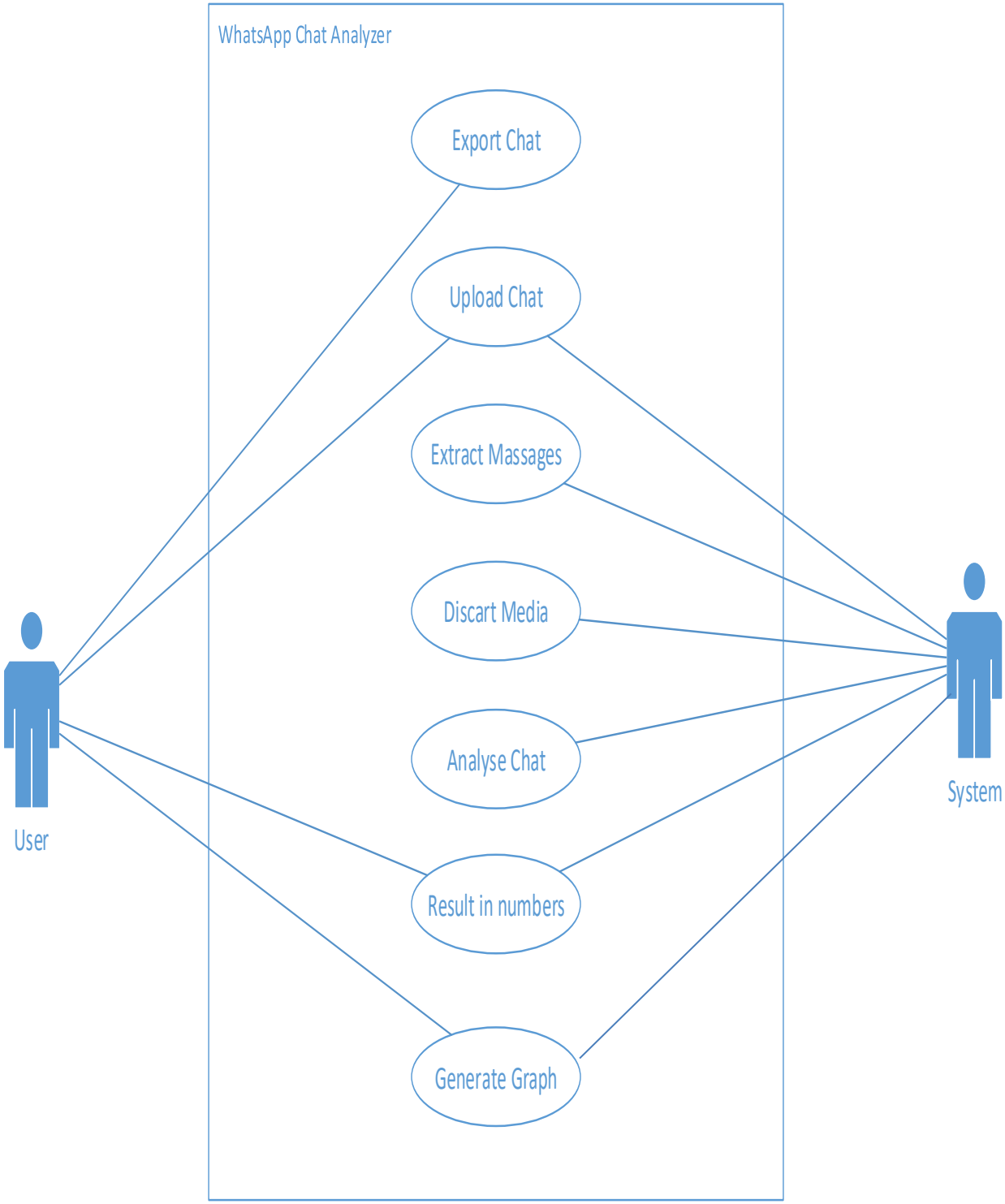
ix. Else increase neutral by 1.

x. Repeat steps 6 and 7 until whole list is analyzed.

xi. Show the result in percentage.

xii. Plot a pie graph of result.

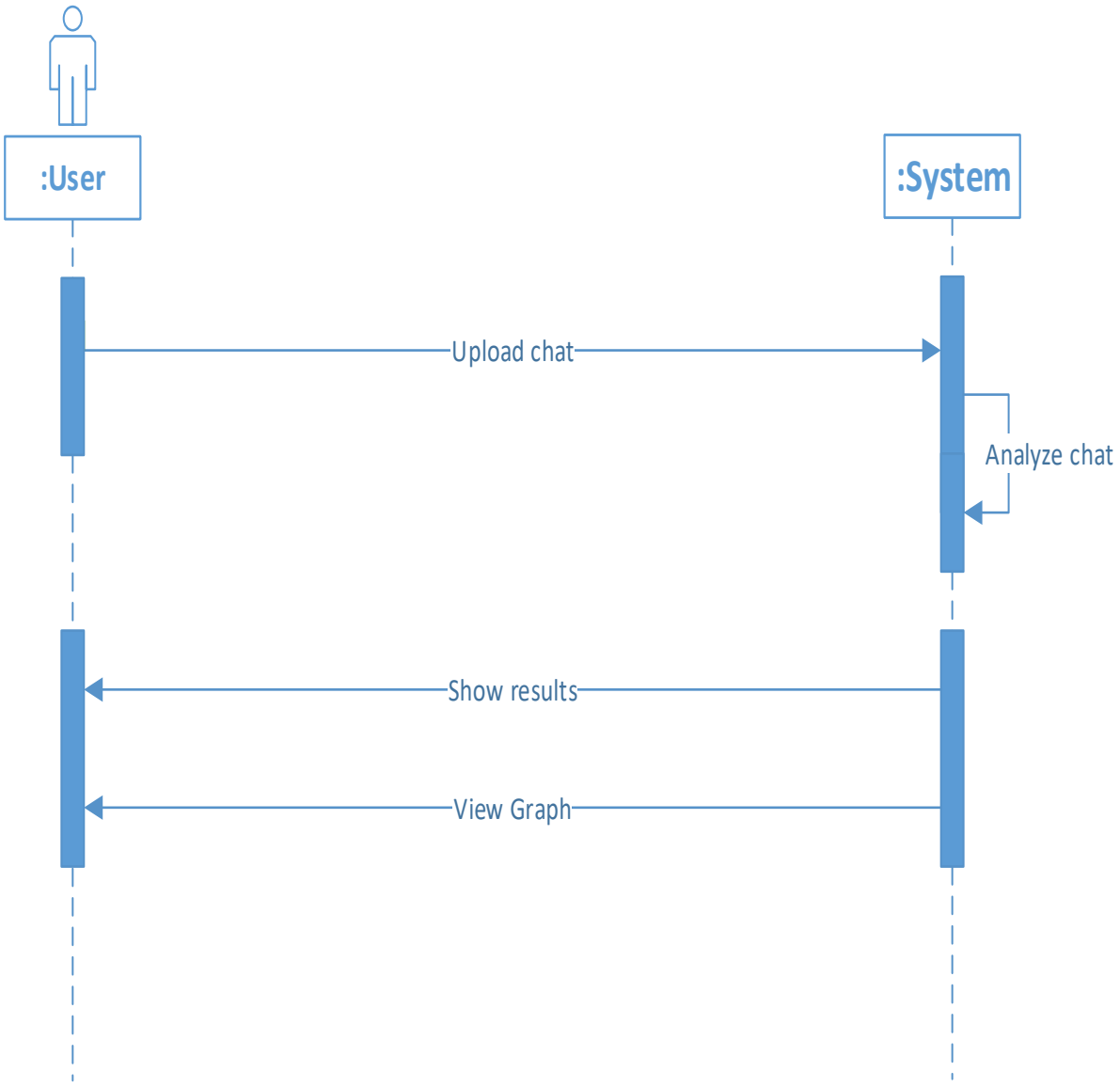
4.3.2 System Activity: Use Case Diagram



- **Use Case Description**:-

The Complete process of WhatsApp chat analysis is as follows, Firstly the user will need to download the chat from the WhatsApp application. This is an in built feature of the application. Then the user needs to send the chat via email or any other medium. After that the user will have to upload the chat into the system which he/she want to analyze. The system will first of all extract the messages from the uploaded chat. As, no sentiment can be derived from images , numbers or time. The system will first discard the images, audio or numerical values from the uploaded chat. We are now left with the data which only contains text values. In python there is a fixed dictionary of the words that are included, there is a fixed priority assigned to each word. Based on the priority the percentage of the sentiments will be affected, for example if the word good has priority of 1 and best has the priority of 3 then the word best will have more impact then good in the final result of the overall positive sentiment of the chat. Finally the whole chat is analyzed this ways and then the result is formed. The result is been shown in the form of pie-chart as well as in the from of percentage, this will enable to have a better view of the chat and desired knowledge can be derived.

4.3.3 Sequence Diagram



- **Sequence diagram description:-**

The process of WhatsApp chat analysis will undergo the following steps. To start with, the user will need to use the inbuilt feature of WhatsApp to download the chat. This is an in built feature of the application. Then the user can transport or send the chat via email or any other medium. After that the user will have to upload the chat into the system which he/she want to analyses. The system will first of all extract the messages from the uploaded chat. As, no sentiment can be derived from images , numbers or time. The system will first discard the images, audio or numerical values from the uploaded chat. We are now left with the data which only contains text values. There is a fixed dictionary or libraries of word of English language in python mainly a good amount of the words are included which will help to get more accurate result. There is a fixed priority assigned to each word. Based on the priority the percentage of the sentiments will be affected, for example if the word bad has priority of 1 and best has the priority of 3 then the word worse will have more impact then bad in the final result of the overall negative sentiment of the chat. Finally the whole chat is analyzed this ways and then the result is formed. The result is been shown in the form of pie-chart as well as in the from of percentage, With the help of this result we will be able to take steps to overcome or maintain the situation.

4.3.4 Limitations & Weakness Of Current System

Input Formats: According to the manual page of NLTK it supports many languages current implementation does not support any kind of local languages like Gujarati , Hindi or any local chat languages.

Supported Languages: Although nltk supports many languages, current implementation supports only English Language. Local handwriting is still an issue with NLTK

Output Formats: Current implementation supports only pie chart as output.

Sarcasm: It is a popular form of mockery to ridicule or convey insult. Analytics fails to recognize these forms of emotions and might prove to be ineffective in such cases. Though the efforts are being made to cater to this problem through the extensive use of machine learning and artificial intelligence and we might see an improved version of sentiment analysis in near future.

For ex: “I am so proud of your stupidity, you make me feel good about myself.”

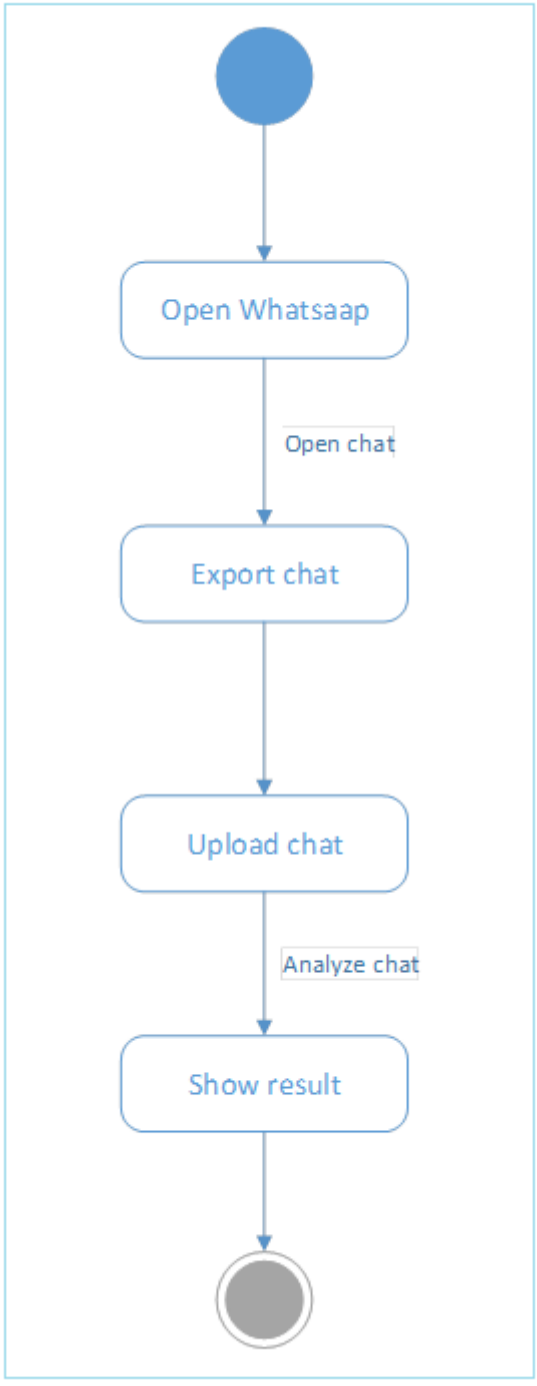
Multiple Meanings: A word could have many meanings and it may represent multiple emotions as we move from one geography to another or even one person to another. Many English words in the UK may mean different in American English.

For ex: “I think you’ve been playing horribly dope.”

Dependency: Sentiment analysis largely depends on the predefined words and their individual score. Which leads to many problems like ambiguity in the context of the sentence? A sentence which includes ‘good’ might not have any emotions attached to it but will be shown as positive by the analysis.

Chapter 5: System Design

5.1 State Transition Diagram

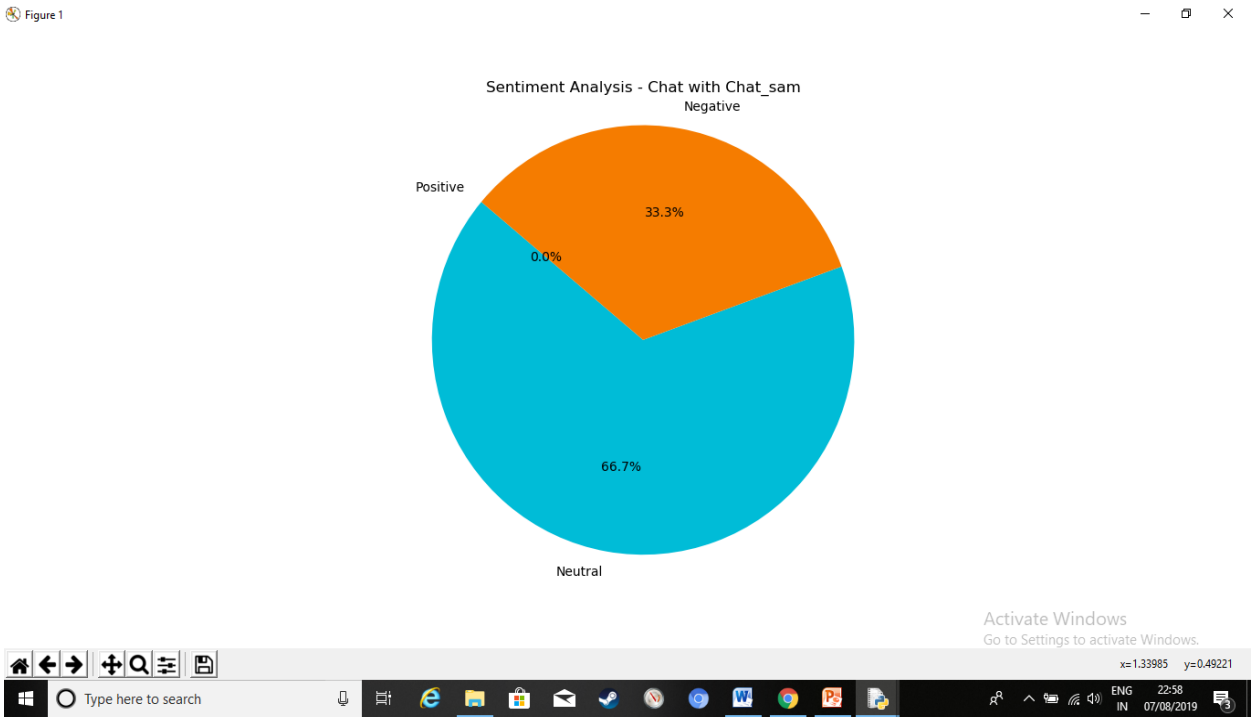


State Diagram Discription:-

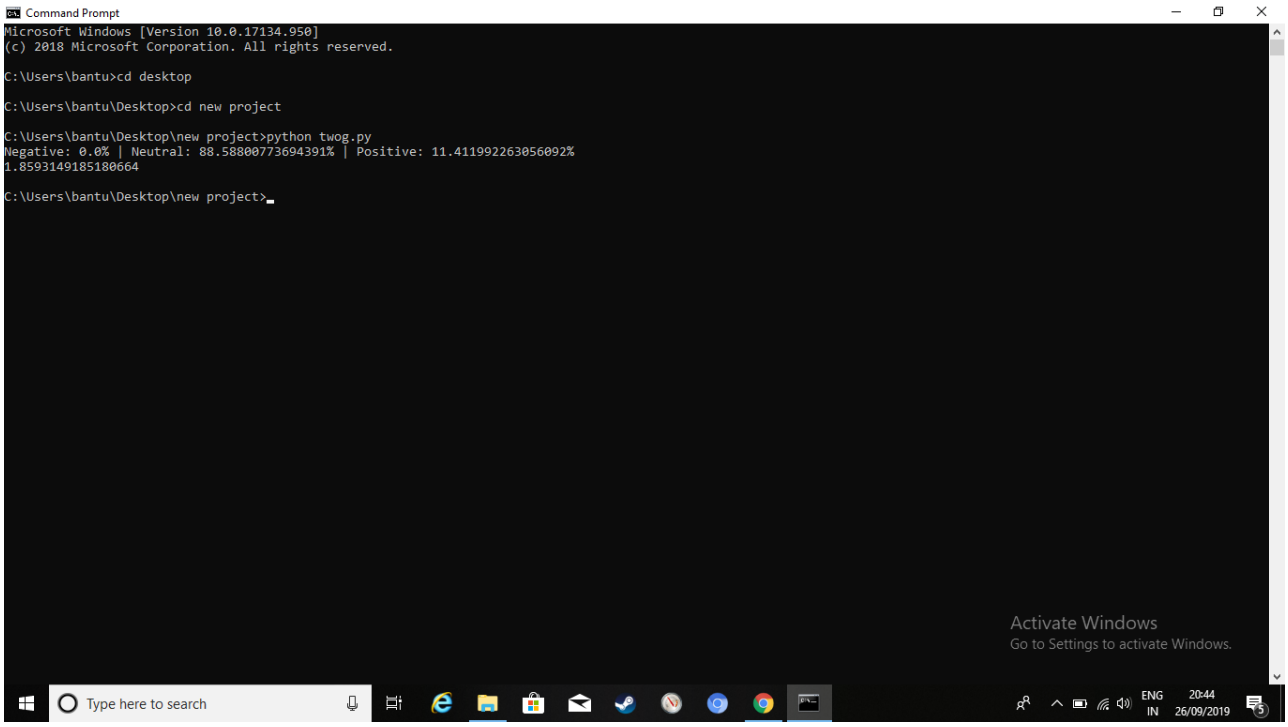
The analysis of watsapp chat will have to follow the following steps. To embark with, the client will have to make sure that its uses the inbuilt feature of whatsapp to download the chat of which heshe wants to find the sentiments of. There is an in built feature of the whatsapp. Then the user can transport or send the chat via email or any other ways that is suitable for him/her. After that the user will have to upload the chat into the system which he/she want to analyse.The chat file is being loaded into the program by its file name. The system will first of all try to extract the messages from the uploaded chat. As, no sentiment can be derieved from images , numbers or time.As a result of it the system will first discard the images, audio or numerical values such as date or time from the uploaded chat. We are now left with the data which only contains text values. There is a fixed dictionary or libraries of word of English language in python mainly a huge amount of the words are included which will help to get more accurate result. No other language is supported by python which is a drawback of it. There is a fixed priority assigned to each word. Based on the priority the percentage of the sentiments will be affected, for example if the word equal has priority of 1 and equilibrium has the priority of 2 then the word equilibrium will have more impact then equal in the final result of the overall neutral sentiment of the chat. Finally the whole chat is analysed this ways and then the result is formed. The result is been shown in the form of pie-chart as well as in the from of percentage, With the help of this result we will be able to gain the information whether the chat is positive, negative or neutral sentiment.

5.2 Output Interface

Output in the form of graph:-



Output in numeric values:-



Chapter 6: TESTING STRATEGY

"Testing cannot show the absence of defect. It can only show that software errors are present."

- Testing is the process of executing a program with the explicit intention of finding errors that is, making the program fail. Testing is very crucial and most expensive phase of the software development. Before delivering the system, the process of rigorous testing is done to check that software works as it is expected and meets its specifications. For that two testing strategies are there Code Testing & Specification Testing. We have used both of them at different levels of code development.

Testing (White Box Testing)

- The code-testing strategy examines the logic of the program. To follow this testing method, test cases should be developed that result in executing every instruction in the program or module; that is, every path through the program is tested. A path is a specific combination of conditions that is handled by the program.

- This testing is used at initial stage of the development, as code volume is very less at this stage. It checks only the aspects are implemented correctly or not.

- But this strategy does not indicate the code meets its specifications nor does it determine weather all aspects are even implemented. So with this, another strategy is also used.

Specification Testing (Black Box Testing)

- In this strategy, the specifications stating what the program should do and how it should perform under various conditions are examined. Test cases are developed for each condition or combination of conditions.
- The analyst does not look into the program to study the code and is not concerned about weather every instruction or path through the program is tested. This is more efficient method, since it focuses on the way software is expected to be used.

6.1 TESTING METHOD

• Different types of testing method are used,

- Unit Testing:-
    - In it analyst tests the program making up a system. The software units in a system are the modules and routines that are assembled and integrated to perform a specific function.
    - It focuses on modules, independently of one another, to locate errors. This enables the tester to detect errors in coding and logic that are contained within the module alone.
- Bottom-Up Unit Testing:-
    - It can be performed from the bottom up, starting with the smallest and lowestlevel modules and proceeding one at a time. For each module in bottom-up testing, a short program executes the module and provides the needed data, so that the module is asked to perform the way it will when embedded within the larger system.
- Top-Down Unit Testing:-
    - As the name implies, begins with the upper-level modules. However, since the detailed activities usually performed in lower-level routines are not provided, stubs are written. A sub is a module can be called by the upper-level module and that, when reached properly, will return a message to the calling module, indicating a proper interaction occurred.
- System Testing:-
    - System testing does not test the software per se but rather the integration of each module in the system. It also tests o find discrepancies between the system and its original objective, current specifications and system documentation.
    - The primary concern is the compatibility of individual modules. Analysts are trying to find areas where modules have been designed with different specifications.

Test cases are designed to test the system and according to the submitted test cases test data are determined and then he system is tested according to different test objectives.

Chapter 7: Conclusion

From a proper analysis of positive points and constraints on the component ,it can be safely concluded that using python and its various toolkit, modules and packages we have built a code which tries to analyze the whatsapp chat text file accurately. And as a result of analysis we display positive ,negative and neutral words in percentage and also display in a pie graph.

Bibliography

- https://www.nltk.org/_modules/nltk/sentiment/vader.html
- https://github.com
- https://matplotlib.org/