WHATSAPP CHAT ANALYZER USING MACHINE LEARNING

Submitted in partial fulfillment of the requirements of Third Year of Engineering in the Department of Computer Engineering, during the academic year 2023-2024.

Submitted By

Group Members Name	Roll Numbers
Vedika Jadhav	514
Sarthak Khochrekar	517
Azeem Ansari	501

Submitted to:

Prof. Diksha V. Kale

(Department of Computer Engineering)

Department Of Computer Engineering

PILLAI COLLEGE OF ENGINEERING

(Autonomous)

New Panvel – 410 206

UNIVERSITY OF MUMBAI

Academic Year 2023-24

Acknowledgement

Efforts have been made and implemented in a unique way with dedication towards completing this project report. For this success, we would like to thank our parents and all our respected teachers for their encouragement. We would like to express our special thanks to Principal Dr. Sandeep Joshi for giving us this platform to showcase our relevant skills. We would like to express our gratitude to The Department of Computer Engineering and our Head of Department Prof. Dr. Sharvari Govilkar for encouraging us throughout and also the Machine Learning Co-ordinator Prof. Diksh V. Kalee who gave us the opportunity to do this mini project as a part of Machine Learning in the topic WhatsApp Chat Analyzer using Machine Learning, which helped us in applying the knowledge that we have acquired during the semester and to learn new concepts and for supporting and helping us throughout the project time period. We would also like to thank the lab assistants and all the librarians for providing us with the latest software and also for helping us with various equipment. We would also like to thank all our colleagues for supporting each other. We are immensely grateful to all of them for sharing their pearls of wisdom with us during this course. We would also like to expand our deepest gratitude to all those who have directly or indirectly guided us in completing this project.

Executive Summary

The most used and efficient method of communication in recent times is an application called WhatsApp. WhatsApp chat analyzer is the application deployed on Heroku web which provide analysis of WhatsApp group. This is the combination of machine learning and NLP. This whatsapp chat analyzer take import whatsapp chat file from user and analyze it and give different visualizations as a result. This tool aims to offer a thorough study of the information that WhatsApp provides. Regardless of the subject around which the conversation is centred, our generated code may be used to improve comprehension of the data. The benefit of this tool is that is implemented using simple python modules such as pandas, matplotlib, seaborn and sentiment Analysis that are used to produce data frames and plot various graphs are then shown in the flutter application. Because this approach is effective and resource-conserving, it can be readily applied to the largest dataset.

TABLE OF CONTENTS

Chapter 1

Introduction

WhatsApp-Analyzer is a statistical analysis tool for WhatsApp chats. Working on the chat files that can be exported from WhatsApp it generates various plots showing, for example, which other participant a user responds to the most. Communication between people using the internet becomes part of their daily life. People used to communicate with each other using the online chat system to transfer their messages. We propose to employ dataset manipulation techniques to have a better understanding of WhatsApp chat present in our phones.It shows most used emoji and word which repeatedly most times. It tracks our conversation and analyzes how much time we are spending. This tool is based on data analysis and processing. The first step in implementing a machine learning algorithm is to understand the right learning experience from which the model starts improving. Data preprocessing plays a major role when it comes to machine learning. In order to make the model more efficient we need lots of data, so we turned our focus primarily on one of the large scale data producers owned by Facebook which is nothing but WhatsApp. WhatsApp claims that nearly 55 billion messages are sent each day. The average user spends 195 minutes per week on WhatsApp, and is a member of plenty of groups. With this treasure house of data right under our very noses, it is imperative that we embark on a mission to gain insights on the messages which our phones are forced to bear witness to. A list that uses pie charts and diagrams to represent the interesting data that it collects after analyzing your WhatsApp chats. You know the drill by now. You will take a backup of your chat and send it to an email id listed on the site.

1.1 Proposed System

The "WhatsApp Chat Analyzer" gives users a platform to analyse WhatsApp talks online via a heroku connection. With this programme, users may explore whatsapp exported (.txt) files, input them into WhatsApp conversation analyzer, and receive analysis based on those txt files. And user can Analyze by clicking Show Analysis button. Applying a sentiment analysis algorithm to exploratory data analysis is the initial stage in this process. This approach offers the positive, negative, and neutral chatter and is used to construct a pie chart depending on these parameters. Plotting a line graph that displays the author and message count for each date, the author and message count for each author, an ordered graph of date against message count, the number of media sent by authors, etc. Show the message with no authors and a graph of the number of messages every hour. Better code representation and user understandability are provided by these diverse modules. Numpy, Scipy Pandas, CSV, Sklearn, Matplotlib, sys, re, emoji, nltk Seaborn, and other libraries are utilised.

1.2 Scope

Data pre-processing, the initial part of the project is to understand implementation and usage of various python- built modules. The above process helps us to understand why different modules are helpful rather than implementing those functions from scratch by the developer. These various

modules provide better code representation and user understandability. The following libraries are used such as numpy, scipy, pandas, csv, sklearn, matplotlib, sys, re, emoji, nltk seaborn etc.

1.3 Methodology

First, a simple working system implementing only a few basic features is built and then that is delivered to the customer. Then thereafter many successive iterations/ versions are implemented and delivered to the customer until the desired system is realized. At any time, the plan is made just for the next increment and not for any kind of long term plans. Therefore, it is easier to modify the version as per the need of the customer. The Development Team first undertakes to develop core features (these do not need services from other features) of the system.

1.4 Limitations

- Maximum file size to be uploaded is 200MB
- Only supports English language
- Support only txt extension

Chapter 2

Literature Review

Ravishankara K, Dhanush, Vaisakh, Srajan I S 1Faculty of Department of Computer Science, Srinivas Institute of Technology Mangalore, Karnataka 2,3Students of Department of Computer Science, Srinivas Institute of Technology Mangalore, Karnataka

The study titled "WhatsApp Chat Analyzer Study: Understanding Communication Dynamics" endeavors to employ the WhatsApp Chat Analyzer, a tool developed by Ravishankara K, Dhanush, Vaisakh, and Srajan I S, to gain comprehensive insights into communication dynamics within WhatsApp groups. The central research question revolves around how the WhatsApp Chat Analyzer can augment our comprehension of WhatsApp chat data and its implications for subsequent analysis. To process and analyze WhatsApp chat data, the study utilized the WhatsApp Chat Analyzer, a Python-based tool. This tool leverages Python modules such as pandas, matplotlib, seaborn, and sentiment analysis to facilitate the creation of data frames, exploratory data analysis, and the generation of visual representations. Through the application of the WhatsApp Chat Analyzer, several significant findings emerged. Notably, June 15, 2018, was identified as the most active date, witnessing the transmission of 190 messages. Additionally, the tool identified the top 10 and top 20 most active users in the chat group, providing percentages of their contributions. The analysis also revealed the word "the" as the most frequently used, appearing a staggering 43,313 times. These results underscore the robust utilization of the WhatsApp platform within the analyzed group, with the high message count on June 15, 2018, hinting at a significant event or discussion. The identification of top contributors sheds light on the group's dynamics, pinpointing individuals who play pivotal roles in the communication flow. Moreover, the prevalence of the word "the" underscores the necessity for employing more advanced text analysis techniques to extract more meaningful insights. In conclusion, this study showcases the WhatsApp Chat Analyzer's effectiveness in providing valuable insights into WhatsApp group communication. The tool's capacity to handle and analyze large datasets allows for a more profound understanding of user behavior and interaction patterns. These findings align with existing literature, highlighting the pivotal role of data-driven insights in the realm of communication platforms.

The literature review delves into studies centered on WhatsApp Chat Analysis employing Machine Learning methodologies. It introduces the WhatsApp Chat Analyzer, a fusion of machine learning and NLP, tailored to conduct in-depth analyses of WhatsApp group conversations. The review seeks to explore the impact, effectiveness, and implications of utilizing WhatsApp as a communication platform, while also highlighting the critical role of high-quality data in training machine learning models for chat analysis. The methods employed across the reviewed studies encompassed a range of techniques for WhatsApp chat analysis. These included data extraction through the 'Export Chat' feature, preprocessing steps like text parsing and stop word elimination, and the application of NLP techniques for text analysis. Python, along with libraries such as Matplotlib, NumPy, and Pandas, played a pivotal role in processing and visualizing the data. Additionally, the research underscored the importance of exploratory data analysis in comprehending various forms of WhatsApp conversations. The findings of the studies were substantial, demonstrating the

widespread popularity of WhatsApp among young users and its extensive usage for communication, media sharing, and diverse other purposes. Moreover, the research highlighted transformative shifts in communication patterns attributed to the pervasive use of social networking sites and smartphones. The analysis of group chats provided valuable insights into user behavior, encompassing message types, response rates, and demographic factors. Forensic analysis shed light on user profiles and system functionalities within the WhatsApp platform. The results underscore the pivotal role of WhatsApp as a communication tool, particularly among the younger demographic, emphasizing the necessity for efficient data analysis techniques to extract meaningful insights from WhatsApp conversations. The studies also spotlight the potential applications of WhatsApp chat analysis in various domains, spanning social sciences, communication studies, and digital forensics. The effectiveness of using Python and related libraries for data processing and visualization was further highlighted through exploratory data analysis.In conclusion, the literature review accentuates the significance of WhatsApp Chat Analysis employing Machine Learning methodologies. It showcases the adaptability and applicability of the WhatsApp Chat Analyzer in conducting comprehensive analyses of group conversations. Collectively, the reviewed studies indicate the potential for broader applications of this technology, not only in understanding communication patterns but also in fields like digital forensics and social sciences. This review lays a solid foundation for further research in this domain.

1Professor, Computer Science and Engineering, A.C.E.T. Nagpur, Maharashtra, India 2B.E. Student, Computer Science and Engineering, A.C.E.T. Nagpur, Maharashtra, India. The WhatsApp Activity Analyzer, developed by a team led by Asst. Prof. Sadia Patka along with students and professionals from A.C.E.T. Nagpur, Maharashtra, India, introduces a robust tool for extracting valuable insights from WhatsApp group chats. This tool employs advanced data analysis techniques to identify key topics, keywords, and message recognition, catering to a wide range of conversations including those in educational, business, and personal spheres. By addressing the challenge of handling extensive chat data, the Analyzer offers substantial potential in enhancing communication strategies and overall productivity. Its user-friendly interface makes it accessible to individuals with basic computer skills. Furthermore, this paper provides a comprehensive overview of the tool's applications, highlighting both its benefits and limitations, ultimately positioning the WhatsApp Activity Analyzer as a valuable resource for anyone seeking to gain meaningful insights from their group conversations. The study is fortified by an extensive literature review encompassing research on WhatsApp's influence on education, communication habits of young users, and its effectiveness in diverse contexts, establishing a solid foundation for the proposed tool's potential impact.

Iduh, B. N. (2020). WhatsApp Network Group Chat Analysis Using Python Programming. International Journal of Latest Technology in Engineering, Management & Applied Science, 9(2).

In the paper titled "WhatsApp Network Group Chat Analysis Using Python Programming" authored by Blessing Nwamaka Iduh, an in-depth exploration of WhatsApp group data analysis is conducted through the utilization of Python programming. The study aims to unveil various facets of user engagement, active dates, message counts, and more within WhatsApp group interactions. Employing Python, alongside essential libraries like Numpy, Pandas, Matplotlib, and Seaborn, the research methodology involves data extraction, transformation, exploration, and visualization. The results shed light on pivotal insights, highlighting the most active date, June 15, 2018, with an impressive count of 190 messages sent. Moreover, a top contributor emerges, having sent over 972 messages, and key statistics such as active administrators, total users, and

individual post counts are revealed. Additionally, the most frequently used word, "the," surfaces a staggering 43,313 times. This analysis not only provides valuable insights into member participation and activity levels but also establishes a robust framework applicable to diverse WhatsApp groups, enabling a deeper comprehension of their dynamics and interactions (Iduh, 2020).

Authors: Aditya Jadhav, Suraj Patil, Mujtaba Shaikh, Prajwal Pal, Prof. Aparna Sawant, Prof. Pankaj Kunekar

The papers discussed in this literature review focus on the analysis of WhatsApp group chats, a pivotal form of contemporary communication. WhatsApp has emerged as a platform where individuals share personal thoughts, sentiments, and engage in group discussions. Understanding the dynamics of these conversations holds significant value in comprehending human interactions in the digital age. The WhatsApp Chat Sentiment Analyzer is an innovative tool that employs a combination of Natural Language Processing (NLP), specifically the VADER (Valence Aware Dictionary for Sentiment Reasoning) library, and machine learning techniques. The system processes exported chat files, applying techniques from libraries such as seaborn, streamlit, numpy, matplotlib, and pandas. Additionally, data preprocessing plays a crucial role in the efficiency of machine learning models, necessitating the utilization of regular expressions (regex) for effective data manipulation. The results obtained from the WhatsApp Chat Sentiment Analyzer offer valuable insights into the group dynamics of WhatsApp conversations. By employing sentiment analysis and machine learning techniques, the system successfully deciphers the underlying sentiments within these conversations. The visual representations provide an intuitive understanding of communication patterns, active users, and frequently used words and emojis. These findings have implications for understanding social interactions in digital spaces and can be valuable for various applications, including social research, marketing, and community management. The WhatsApp Chat Sentiment Analyzer is a significant contribution to the study of digital communication. By harnessing the power of NLP and machine learning, it provides a robust framework for analyzing WhatsApp group chats, accessible to a wide range of users through streamlined Python libraries. Moving forward, further research in this area could explore the integration of advanced NLP techniques and expand the scope to analyze conversations in different social media platforms, providing a deeper understanding of digital interactions.

Journal of Emerging Technologies and Innovative Research (JETIR)

This paper delves into an extensive analysis of WhatsApp conversations, specifically focusing on the identification of spam and legitimate messages. It employs a range of supervised machine learning algorithms, including naïve Bayes, support vector machines, and the maximum entropy algorithm, evaluating their effectiveness in differentiating between ham and spam messages. Emphasizing the diverse nature of discussions in WhatsApp groups, the study highlights the prevalent issue of SMS spamming, which often leads to users missing out on crucial messages amidst the inundation of unwanted content. The authors aim to develop a classification algorithm tailored for SMS spam filtering, taking into account the unique challenges posed by this form of communication, such as limited character space and the use of shorthand and slang. Their methodology involves utilizing the publicly available SMS Spam collection

dataset, consisting of 5574 records, and implementing preprocessing steps to prepare the text messages for analysis. Through visualization techniques like word clouds, the study identifies key differences in word frequencies between ham and spam messages, shedding light on potential keywords for classification. The research yields noteworthy findings, showcasing the varying accuracies of different classification algorithms. The SVM algorithm emerges as the most precise, achieving up to 98% accuracy, followed by the naïve Bayes algorithm, and then the maximum entropy algorithm. The study underscores the complexity of distinguishing between ham and spam messages, particularly within the constraints of SMS communication. Ultimately, the findings hold significance for mobile phone providers seeking to enhance user experience by implementing robust spam filters. However, the study calls for further research and refinement of algorithms to adapt to the evolving landscape of SMS communication and spamming techniques.

Chapter 3

Requirement Analysis

Requirements Analysis is the process of defining the expectations of the users for an application that is to be built or modified. Requirement analysis involves all the tasks that are conducted to identify the needs of different stakeholders.

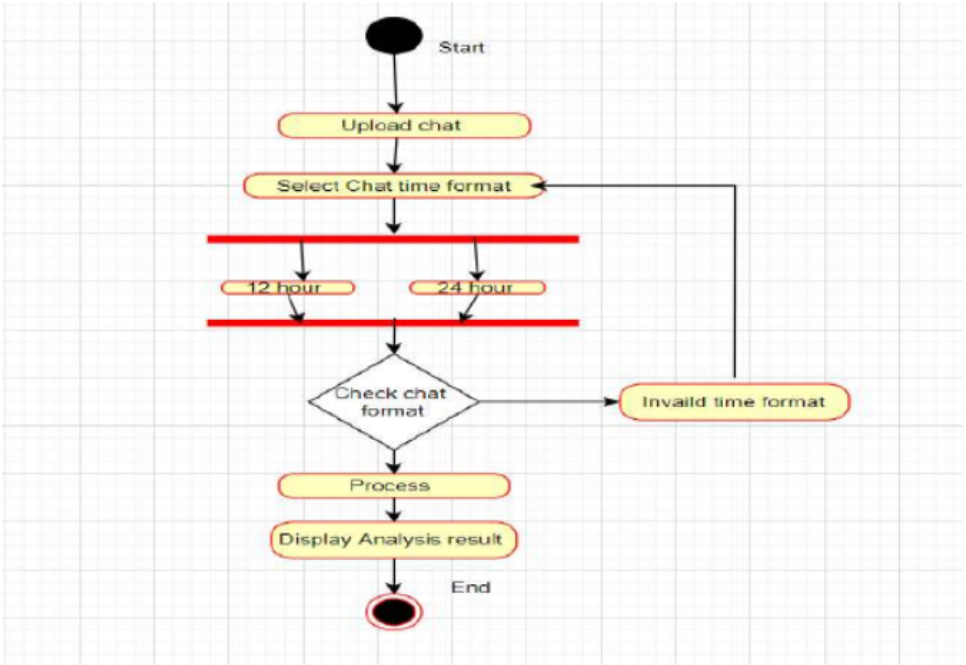
3.1 System Requirements

Software requirement for developing application

- Jupyter notebook
- Python and its libraries
- ML algorithm

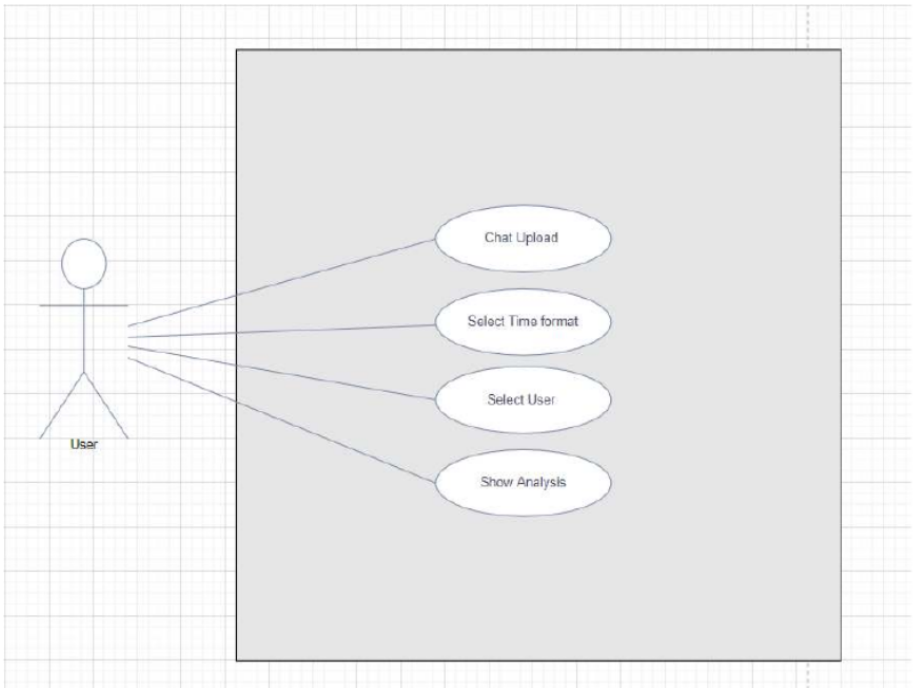
3.2 Activity Diagram

●In the activity diagram as the initial activity starts user will upload the file as input which is action and in the next action time format will be selected.
●The decision box check chat format represents the validity of the time format of the file.
●If the time format is correct then analysis will be done and process will end.
●If the time format is wrong user will have to again check for the correct format.



3.3 Use Case Diagram

●In the use case diagram the actor is User.
●Users can make use of chat upload use cases to give input to the system.
●Select time format use case describes that user can input the time format of the file in the system.
●Select user use case is to select whose analysis result is desired.
●Users can make use of Show analysis use cases to see the result of the entire analysis done by the system



Chapter 4
Software Description

The "WhatsApp Chat Analyzer" gives users a platform to analyse WhatsApp talks online via a heroku connection. With this programme, users may explore whatsapp exported (.txt) files, input them into WhatsApp conversation analyzer, and receive analysis based on those txt files. And user can Analyze by clicking Show Analysis button. Applying a sentiment analysis algorithm to exploratory data analysis is the initial stage in this process

4.1 Python

Python is a robust programming language that is simple to learn. Its object-oriented programming methodology is straightforward but efficient, and it includes good high-level data structures. Python's interpreted nature, intuitive syntax, and dynamic typing make it the perfect language for scripting and quick application development across a wide range of platforms.

4.2 Libraries & Modules Imported

- **Streamlit**: Streamlit is a free and open-source python framework. [2] We can quickly develop web apps for Machine Learning and Data Science by using Streamlit. Streamlit can easily integrates with other popular python packages such as NumPy, Pandas, Matplotlib, Seaborn. Streamlit provides fastest way to develop and deploy web apps.

- **Matplotlib**: Matplotlib is a popular Python packages used for data visualization. It is a cross-platform library for making plots from data in arrays. It helps in creating static, animated and interactive visualizations in python.

- **Seaborn**: Seaborn is the data visualization library. It is used for making statistical graphs. Visualization is the central part of seaborn. Seaborn provides exploration and better understanding of data. Seaborn closely integrates into the data structures from python.

- **Pandas**:
  Pandas is an open-source python library. Pandas used to convert string data into Data frame. Data frame is the representation of data into 2-dimensional table of rows and columns. We can work with large data sets using Pandas library. Pandas library has many built-in functions for data analysis, data cleaning, data exploration and data manipulation

- **Support Vector Machine(SVM)**:
  Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is

used for Classification problems in Machine Learning.The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.

- **URLExtract**:
  A python class for collecting (extracting) URLs from given text based on locating TLD.

- **Word Cloud**:
  A data visualization technique used for representing text data in which the size of each word indicates its frequency or importance.

- **Collections**:
  Containers used for storing data and are commonly known as data structures, such as lists, tuples, arrays, dictionaries, etc.

- **VaderSentiment**:
  VADER (Valence Aware Dictionary and sEntiment Reasoner) is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media, and works well on texts from other domains.

- **Scikit-learn (Sklearn)**:
  It is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib.

Chapter 5

Project Implementation

Analyzing WhatsApp chat data can be a valuable way to gain insights into your messaging patterns, communication trends, and even sentiment over time. Here's a basic outline of how you can implement WhatsApp chat analysis:

Note: To perform this analysis, you will need to export your WhatsApp chat data, which you can do within the WhatsApp app. The steps to export chat data may vary slightly depending on the operating system (iOS or Android) and WhatsApp version.

Step 1: Export WhatsApp Chat Data

1. Open the WhatsApp conversation you want to analyze.
2. Tap on the contact's name at the top of the chat.
3. Scroll down and select "Export Chat."
4. Choose whether to include media files (images, videos, etc.) or not.
5. Select the sharing method (e.g., send to email or save to cloud storage).
6. Export the chat as a text file.

Step 2: Prepare the Data

Once you've exported the chat data, you'll need to prepare it for analysis. You can do this by:

- Saving the exported chat file in a location on your computer.
- Converting the text file into a format that's suitable for analysis, such as a CSV or JSON file.

Step 3: Data Analysis

Here are some analyses you can perform on your WhatsApp chat data:

1. Basic Statistics:
   - Count the number of messages sent and received.
   - Calculate the average message length.
   - Determine the most active days and times of communication.
2. Word Frequency Analysis:

- Count the frequency of words used in the chat.
- Identify the most commonly used words or phrases.
- Create word clouds or frequency plots.

3. Sentiment Analysis:
- Analyze the sentiment of messages to understand the emotional tone of the conversation.
- You can use sentiment analysis libraries like NLTK or VADER for this purpose.

4. Emoticon and Emoji Analysis:
- Count the number of emoticons and emojis used.
- Identify the most frequently used ones.

5. Conversation Trends:
- Plot a timeline of the conversation to see how it has evolved over time.
- Identify any recurring themes or topics.

6. Network Analysis:
- If it's a group chat, you can perform network analysis to see who interacts with whom the most.
- Visualize the social network within the group.

7. Message Frequency Analysis:
- Analyze the frequency of messages sent by each participant.
- Determine if there are any significant patterns or changes in participation.

Step 4: Visualization

To make the analysis more understandable and visually appealing, you can create various charts, graphs, and visualizations using tools like Python (Matplotlib, Seaborn, Plotly), R, or data visualization platforms like Tableau or Power BI.

Step 5: Interpretation

Interpret the results of your analysis. What insights have you gained from the data? Do you notice any interesting trends or patterns in your chat history?

WhatsApp chat analysis can provide insights into your communication habits, help you identify trends in your conversations, and even offer a historical record of your interactions. The specific tools and methods you use for analysis will depend on your goals and the

5.1 Algorithm

Analyzing WhatsApp chat data involves several steps, including data preprocessing, text processing, and various analyses. Below, I'll outline an algorithm for WhatsApp chat analysis:

Step 1: Data Preprocessing

- Input: WhatsApp chat data in a text format (exported from WhatsApp).

- Output: Preprocessed data ready for analysis.

1. Read the chat data from the text file.
2. Extract the relevant information, such as sender, message content, and timestamp.
3. Remove any metadata or system-generated messages.
4. Parse the timestamps into a standardized format.

Step 2: Text Preprocessing

- Input: Preprocessed chat data.

- Output: Cleaned and tokenized text data.

1. Tokenize the text data (split messages into words or tokens).
2. Convert text to lowercase to make it case-insensitive.
3. Remove stop words (common words like "the," "is," "and").
4. Perform stemming or lemmatization to reduce words to their root form.
5. Remove special characters, numbers, and any irrelevant symbols.
6. Create a bag of words or TF-IDF representation for the messages.

Step 3: Basic Statistics

- Input: Cleaned and tokenized chat data.

- Output: Basic statistics about the chat.

1. Calculate the total number of messages sent and received.
2. Compute the average message length (in terms of words).
3. Identify the most active participants (by the number of messages sent).
4. Determine the most active days and hours for communication.

Step 4: Word Frequency Analysis

- Input: Tokenized and preprocessed chat data.

- Output: Word frequency analysis results.

1. Count the frequency of words in the chat.

2. Identify the most commonly used words or phrases.
3. Create a word cloud or frequency plot for visualization.

Step 5: Sentiment Analysis

- Input: Cleaned chat data.

- Output: Sentiment analysis results.

1. Perform sentiment analysis on the messages to determine their emotional tone (positive, negative, neutral).
2. Calculate sentiment scores for the entire conversation or specific participants.

Step 6: Emoticon and Emoji Analysis

- Input: Cleaned chat data.

- Output: Emoticon and emoji analysis results.

1. Count the number of emoticons and emojis used in the chat.
2. Identify the most frequently used emoticons and emojis.

Step 7: Conversation Trends

- Input: Preprocessed chat data.

- Output: Visualization of conversation trends.

1. Plot a timeline of the conversation to see how it has evolved over time.
2. Identify recurring themes or topics over different time periods.

Step 8: Network Analysis

- Input: Chat data from group chats.

- Output: Network analysis results.

1. Analyze the interactions within the group to create a social network graph.
2. Identify central participants and subgroups within the chat.

Step 9: Message Frequency Analysis

- Input: Chat data.

- Output: Analysis of message frequency.

1. Analyze the frequency of messages sent by each participant.
2. Identify patterns or changes in participation.

Step 10: Visualization and Interpretation

- Input: Results from the analysis.

- Output: Visualization and insights.

1. Create various charts, graphs, and visualizations to represent the results.
2. Interpret the analysis results to gain insights into your chat data.

5.2 System Overview

A system overview of WhatsApp chat analysis involves understanding the components and processes required to perform this analysis. Below is a high-level system overview of how a WhatsApp chat analysis system might be structured:

1. Data Collection:

- Input: WhatsApp chat data files exported from the application.

- Process: Users export their chat data and provide the exported files to the system.

- Output: A collection of chat data files for analysis.

2. Data Preprocessing:

- Input: Raw chat data files.

- Process:

    - Parse the data to extract relevant information (e.g., sender, message content, timestamp).

    - Remove metadata and system-generated messages.

    - Standardize timestamps.

- Output: Preprocessed chat data ready for analysis.

3. Text Preprocessing:

- Input: Preprocessed chat data.
- Process:
    - Tokenize the text (split messages into words or tokens).
    - Convert text to lowercase for case-insensitivity.
    - Remove stop words, special characters, and numbers.
    - Apply stemming or lemmatization to reduce words to their root form.
    - Create a bag of words or TF-IDF representation for the messages.
- Output: Cleaned and tokenized text data for analysis.

4. Analysis Engine:

- Input: Preprocessed and tokenized chat data.
- Process:
    - Conduct various analyses on the data, including:
    - Basic statistics (e.g., message count, average message length).
    - Word frequency analysis.
    - Sentiment analysis.
    - Emoticon and emoji analysis.
    - Conversation trends and timelines.
    - Network analysis (for group chats).
    - Message frequency analysis.
- Output: Analysis results, including statistics and insights.

5. Visualization Engine:

- Input: Results from the analysis.
- Process:
    - Create charts, graphs, and visualizations to represent the analysis results.
    - Generate word clouds, sentiment plots, network graphs, and other visual aids.

- Output: Visual representations of analysis results.

6. User Interface:
   - Input: Visualizations and analysis results.
   - Process:
     - Display the results to the user in a user-friendly interface.
     - Allow users to interact with and explore the analysis.
   - Output: A user interface for accessing and exploring chat analysis.

7. Interpretation and Insights:
   - Input: Analysis results and visualizations.
   - Process:
     - Provide insights and summaries based on the analysis.
     - Highlight patterns, trends, and noteworthy findings.
   - Output: Insights and summaries for the user.

8. Export and Sharing:
   - Input: User requests.
   - Process:
     - Allow users to export analysis results, visualizations, and insights.
     - Support sharing analysis findings with others, including through reports or data exports.
   - Output: Exported analysis results and shared insights.

9. Data Storage and Management:
   - Input: Chat data, analysis results.
   - Process:
     - Store chat data and analysis results for future reference.

- Ensure data security and privacy.

- Output: Persistent data storage and management.

10. User Authentication and Authorization:

- Input: User access requests.

- Process:

    - Authenticate users to ensure access to their own chat data.

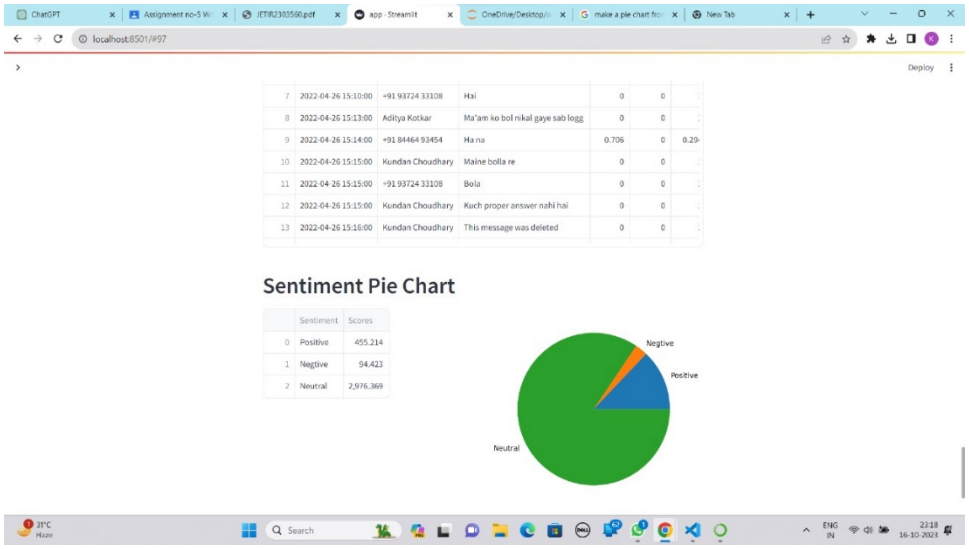
    - Implement access controls and user roles for privacy and security.

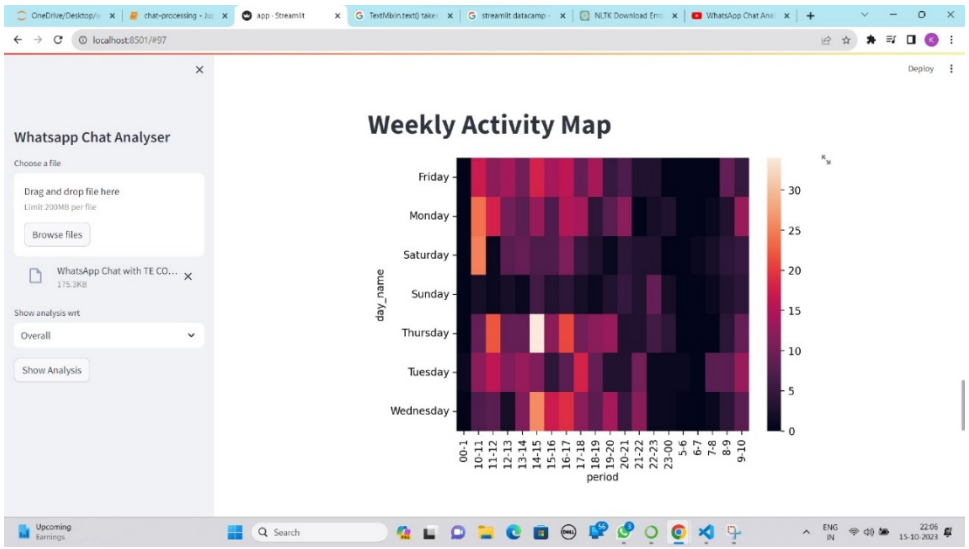
- Output: Authorized access to chat data and analysis.

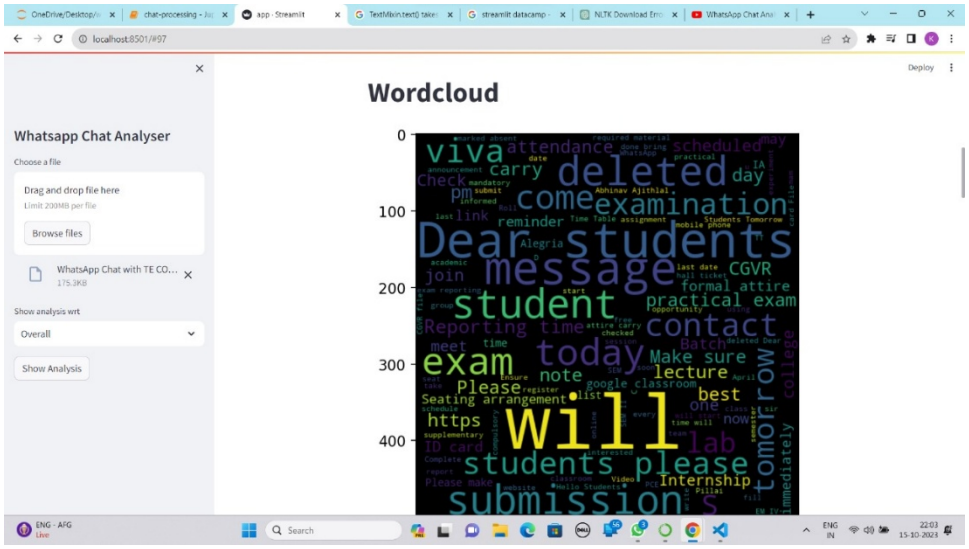
This system overview outlines the major components and processes involved in WhatsApp chat analysis, from data collection to insights generation and sharing. The system can be implemented as a web application, desktop software, or integrated into a larger data analytics platform depending on the requirements and use cases.

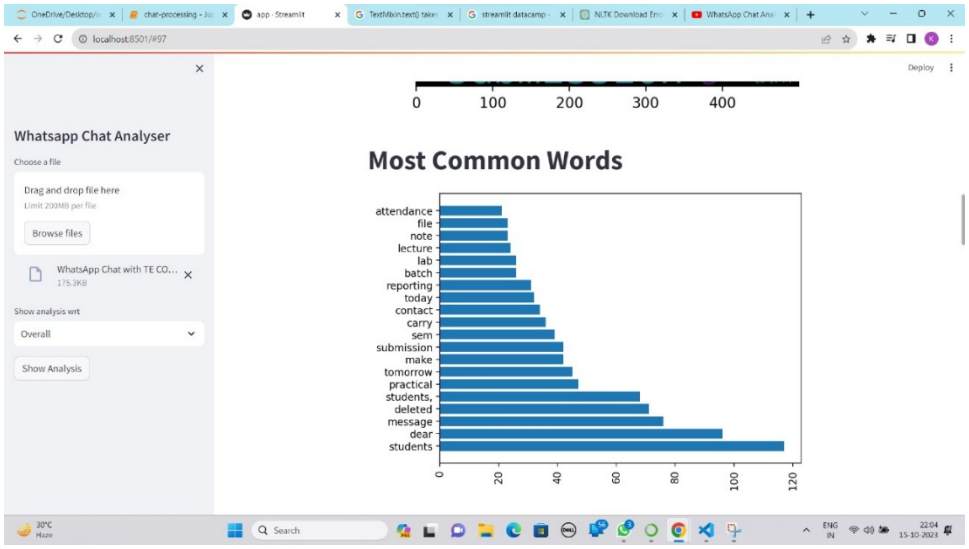
Chapter 6

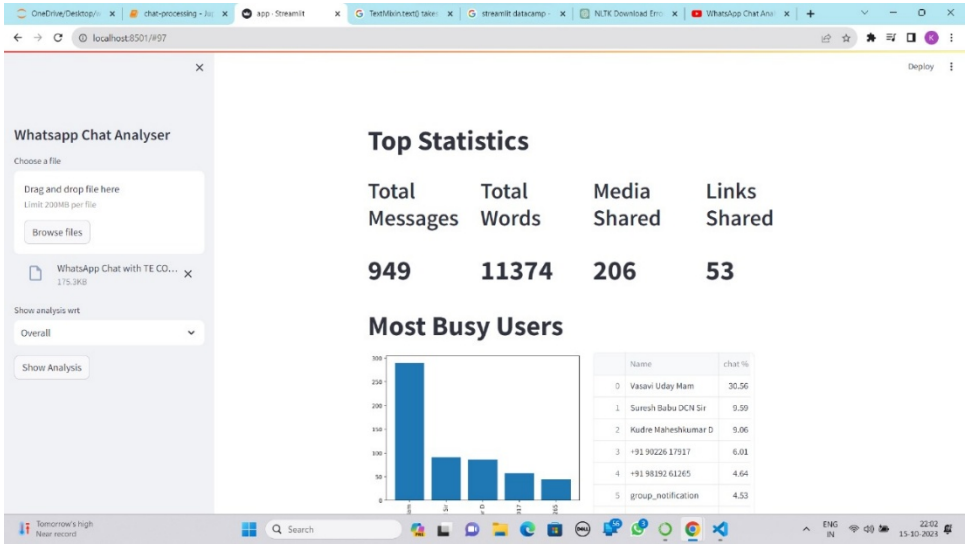
Result and Output

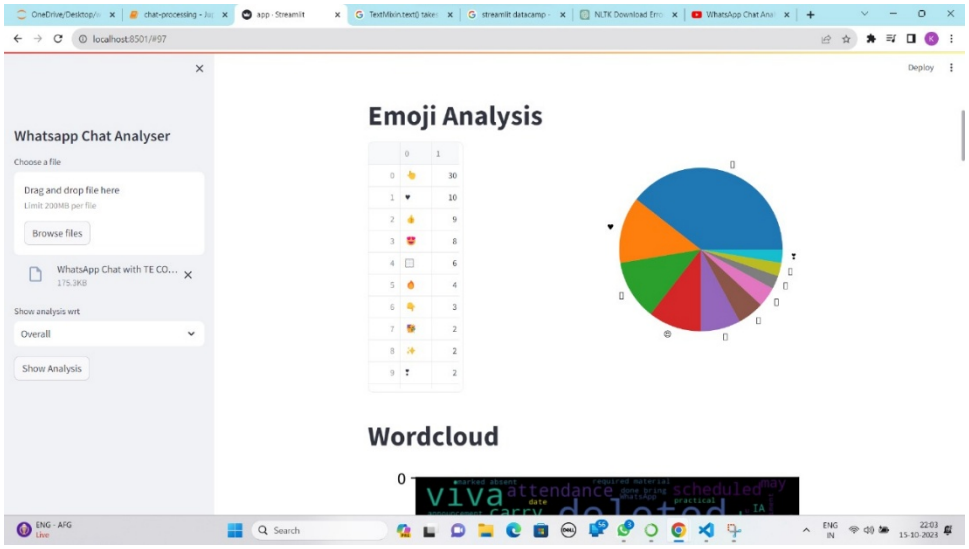


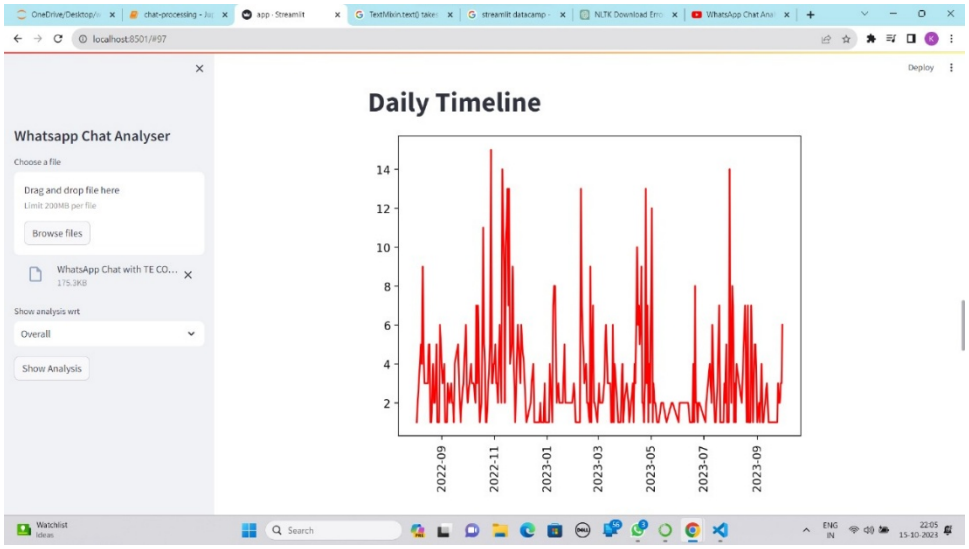


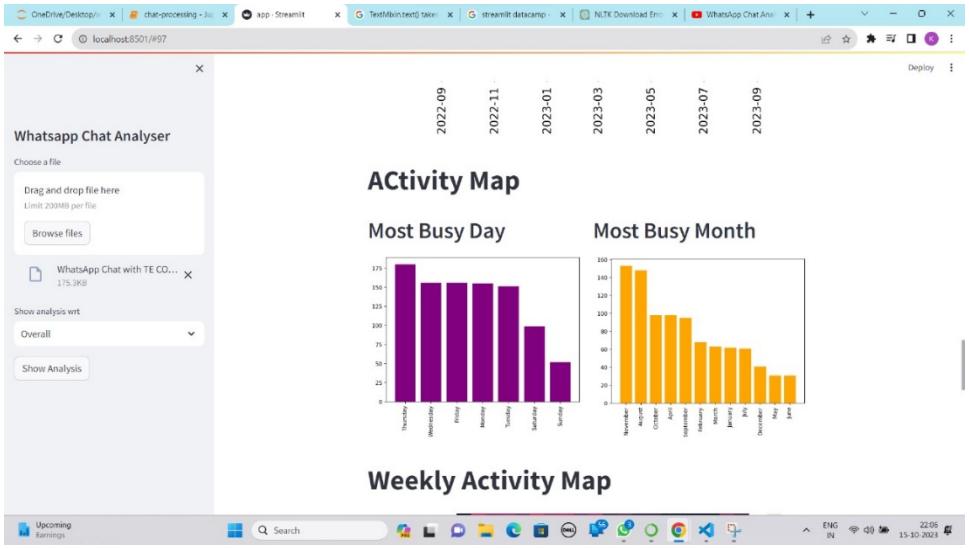

Whatsapp Chat Analyser
Choose a file
Drag and drop file here
Limit 200MB per file
Browse files
WhatsApp Chat with TE CO...
175.3KB
Show analysis wrt
Overall
Show Analysis
Deploy
Wordcloud
0
100
200
300
400
viva
attendance
deleted
come
examination
Dear
students
message
student
exam
today
will
contact
tomorrow
Make sure
lecture
Please
note
Seating arrangement
https
lab
students please
Internship
submission
practical exam
formal attire
CGVR
best

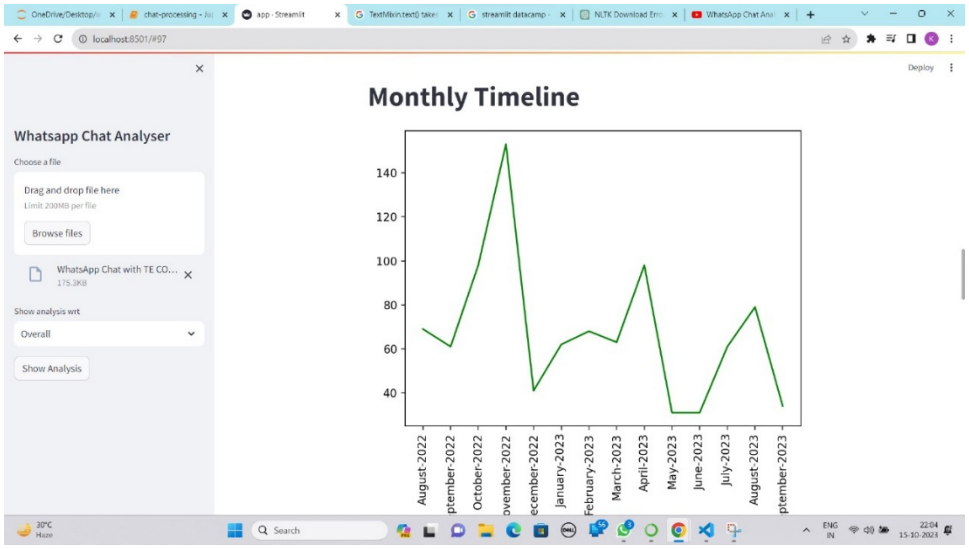

Whatsapp Chat Analyser
Choose a file
Drag and drop file here
Limit 200MB per file
Browse files
WhatsApp Chat with TE CO...
175.3KB
Show analysis wrt
Overall
Show Analysis
Deploy
0
100
200
300
400
Most Common Words
attendance
file
note
lecture
lab
batch
reporting
today
contact
carry
sem
submission
make
tomorrow
practical
students,
deleted
message
dear
students
0
20
40
60
80
100
120


Whatsapp Chat Analyser
Choose a file
Drag and drop file here
Limit 200MB per file
Browse files
WhatsApp Chat with TE CO...
175.3KB
Show analysis wrt
Overall
Show Analysis
Deploy
Top Statistics
Total Messages
949
Total Words
11374
Media Shared
206
Links Shared
53
Most Busy Users
Name
chat %
0 Vasavi Uday Mam 30.56
1 Suresh Babu DCN Sir 9.59
2 Kudre Maheshkumar D 9.06
3 +91 90226 17917 6.01
4 +91 98192 61265 4.64
5 group_notification 4.53


Whatsapp Chat Analyser
Choose a file
Drag and drop file here
Limit 200MB per file
Browse files
WhatsApp Chat with TE CO...
175.3KB
Show analysis wrt
Overall
Show Analysis
Deploy
Emoji Analysis
Wordcloud
viva
attendance
scheduled


Whatsapp Chat Analyser
Choose a file
Drag and drop file here
Limit 200MB per file
Browse files
WhatsApp Chat with TE CO...
175.3KB
Show analysis wrt
Overall
Show Analysis
Deploy
Daily Timeline
2022-09
2022-11
2023-01
2023-03
2023-05
2023-07
2023-09


Whatsapp Chat Analyser
Choose a file
Drag and drop file here
Limit 200MB per file
Browse files
WhatsApp Chat with TE CO...
175.3KB
Show analysis wrt
Overall
Show Analysis
Deploy
2022-09
2022-11
2023-01
2023-03
2023-05
2023-07
2023-09
ACtivity Map
Most Busy Day
Most Busy Month
Weekly Activity Map


Whatsapp Chat Analyser
Choose a file
Drag and drop file here
Limit 200MB per file
Browse files
WhatsApp Chat with TE CO...
175.3KB
Show analysis wrt
Overall
Show Analysis
Deploy
Monthly Timeline
140
120
100
80
60
40
August-2022
ptember-2022
October-2022
ovember-2022
ecember-2022
January-2023
February-2023
March-2023
April-2023
May-2023
June-2023
July-2023
August-2023
ptember-2023

Chapter 7

Conclusion

7.1 Conclusion

In conclusion, it can be said that the capabilities of the WhatsApp application and the power of the python programming language in implementing whatever data analysis intended, cannot be overemphasized. This work was to discuss the WhatsApp application and python libraries, to create an analysis of a WhatsApp chat .We propose to employ dataset manipulation techniques to have a better understanding of WhatsApp chat present in our phones.It shows most used emoji and word which repeatedly most times. It tracks our conversation and analyzes how much time we are spending.The system was done with python, and the python libraries that were implemented includes, NumPy, Pandas, Matplotlib and Seaborn. At the end of the work expected results were obtained and the analysis was able to show the level of participation of the various individuals on the given WhatsApp group chat . On serious note this system has the ability to analyze any WhatsApp chat input into it.

7.2 Future Scope

A The future scope for WhatsApp chat analysis is promising, with opportunities for innovation and expansion in several areas. Here are some key directions where WhatsApp chat analysis can evolve in the future:

1. Advanced Sentiment Analysis: Enhancing sentiment analysis to understand not just positive, negative, or neutral sentiment but also to recognize nuanced emotions, sarcasm, and cultural context.

2. Language and Multilingual Analysis: Developing more robust language models and chat analysis tools that can handle multiple languages and dialects. This can facilitate cross-cultural communication analysis.

3. Contextual Analysis: Analyzing chat conversations in the context of external factors, such as news events, social trends, or even environmental factors. This could provide a more comprehensive understanding of conversation dynamics.

4. Deep Learning and NLP: Leveraging deep learning techniques to create more accurate and context-aware chat analysis models, including improved chatbots and AI-powered assistants for real-time analysis.

5. Real-time Analysis: Implementing real-time chat analysis to provide immediate feedback on conversations, which could have applications in customer service, mental health support, and more.

6. Privacy-Preserving Analysis: Developing techniques for analyzing chat data while preserving user privacy. This is especially relevant as privacy concerns continue to grow.

7. Chatbot Development: Integrating chat analysis into the development of more I telligent and context-aware chatbots for better automated responses.

8. Cross-Platform Integration: Extending chat analysis to work across multiple messaging platforms, allowing users to analyze conversations from various sources in one place.

The future scope for WhatsApp chat analysis is vast and evolving, driven by advances in natural language processing (NLP), machine learning, and user demand for better insights into their communication patterns. However, it is essential to balance innovation with privacy and security concerns to ensure that chat analysis remains ethical and respectful of user rights and data protection regulations.

References

- Available from: http://www.statista.com/statistics/260819/number-of-monthly-active-WhatsApp-users. Number of monthly active WhatsApp users worldwide from April 2013 to February 2016(in millions).
- Ahmed, I., Fiaz, T., "Mobile phone to youngsters: Necessity or addiction", African Journal of Business Management Vol.5 (32), pp. 12512-12519, Aijaz, K. (2011).
- Aharony, N., T., G., The Importance of the WhatsApp Family Group: An Exploratory Analysis. "Aslib Journal of Information Management, Vol. 68, Issue 2, pp.1-37" (2016).
- Access Data Corporation. FTK Imager, 2013. Available at http://www.accessdata.com/support/product- downloads.
- D.Radha, R. Jayaparvathy, D. Yamini, "Analysis on Social Media Addiction using Data Mining Technique", International Journal of Computer Applications (0975 – 8887) Volume 139 – No.7, pp. 23- 26, April 2016.
- Jessica Ho, Ping Ji, Weifang Chen, Raymond Hsieh, "Identifying google talk", IEEE International Conference on Intelligence and Security Informatics, ISI '09, pp. 285-290, 2009.
- Mike Dickson, "An examination into AOL instant messenger 5.5 contact identification.", Digital Investigation, ScienceDirect, vol. 3, issue 4, pp. 227-237, 2006.
- WhatsApp, "About WhatsApp," 20 April 2019. [Online]. Available: https://www.whatsapp.com/about/.
- W. Bani, "Whatsapp," 22 April 2019. [Online]. Available: https://en.welibani.org/weli/WhatsApp. [Accessed 24 April 2019].
- S. Patil, "WhatsApp Group Data Analysis with R," International Journal of Computer Applications,
- https://www.researchgate.net/publication/341876645_Whatsapp_Chat_Analyzer

Appendix

---

1. How easy was it to use our web application?

   - Š Very easy
   - Š Moderately easy
   - Š Slightly easy
   - Š Not at all

2. How user-friendly is the website?

   - Š Very user-friendly
   - Š Moderately user-friendly
   - Š Slightly user-friendly
   - Š Not at all

3. Is the website appearing on your device correctly?

   - Š Yes
   - Š No
   - Š Yes, but with some issues. (Please, specify your screen size)

4. Are you satisfied with the speed and performance of our application?

   - Š Very satisfied
   - Š Moderately satisfied
   - Š Slightly satisfied
   - Š Not at all

Index