WHATSAPP CHAT ANALYZER

A PROJECT REPORT

Submitted by

PRETI SHERINE P (210701195)
YUVAMALINI M (210701507)

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

ANNA UNIVERSITY, CHENNAI

MAY 2024

RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

BONAFIDE CERTIFICATE

Certified that this Thesis titled **“WHATSAPP CHAT ANALYZER**” is the Bonafide work of “**PRETI SHERINE P (2116210701195), YUVAMALINI M (2116210701507)**” who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Dr. RAKESH KUMAR M, M.E., Ph.D.,

PROJECT COORDINATOR

Assistant Professor (SG)

Department of Computer Science and Engineering

Rajalakshmi Engineering College

Chennai - 602 105

Submitted to Project Viva-Voce Examination held on___________________

Internal Examiner **External Examine**r

ABSTRACT

The proposed WhatsApp Chat Analyzer is a user-friendly tool designed to simplify and enhance the analysis of chat conversations on the platform. It automates the process of extracting valuable insights from chat data, offering detailed metrics such as message count, word frequency, media sharing patterns, and emoji usage. The tool provides intuitive visualizations like word clouds and activity timelines to help users understand communication patterns more effectively. With features for sentiment analysis and identification of common phrases, the abstract aims to highlight the system's ability to empower users in making data-driven decisions and improving their communication skills on WhatsApp.[redacted]The WhatsApp Chat Analyzer is envisioned as a comprehensive solution for users seeking deeper insights into their chat conversations. By automating the analysis process, it simplifies the extraction of valuable metrics such as message count, word frequency, media sharing patterns, and emoji usage. This tool goes beyond basic analytics by offering intuitive visualizations like word clouds and activity timelines, providing users with a clear understanding of their communication habits and trends. With additional features for sentiment analysis and identification of common phrases, the system aims to empower users to make informed decisions and enhance their interactions on the platform. Ultimately, the WhatsApp Chat Analyzer represents a significant advancement in chat analysis technology, catering to both personal and professional users alike.

ACKNOWLEDGMENT

First, we thank the almighty god for the successful completion of the project. Our sincere thanks to our chairman **Mr. S. Meganathan B.E., F.I.E.,** for his sincere endeavor in educating us in his premier institution. We would like to express our deep gratitude to our beloved Chairperson **Dr. Thangam Meganathan Ph.D.,** for her enthusiastic motivation which inspired us a lot in completing this project and Vice Chairman **Mr. Abhay Shankar Meganathan B.E., M.S.,** for providing us with the requisite infrastructure.

We also express our sincere gratitude to our college Principal, **Dr. S. N. Murugesan M.E., PhD.,** and **Dr. P. KUMAR M.E., PhD, Director computing and information science , and Head Of Department of Computer Science and Engineering** and our project coordinator **Dr.RAKESH KUMAR M, M.E., Ph.D** for his encouragement and guiding us throughout the project towards successful completion of this project and to our parents, friends, all faculty member sand supporting staffs for their direct and indirect involvement in successful completion of the project for their encouragement and support.

PRETI SHERINE P

YUVAMALINI M

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	**iii**
	LIST OF TABLES	**v**
	LIST OF FIGURES	**vii**
1.	**INTRODUCTION**	**1**
	1.1 RESEARCH PROBLEM	
	1.2 PROBLEM STATEMENT	
	1.3 SCOPE OF THE WORK	
	1.4 AIM AND OBJECTIVES OF THE PROJECT	
	1.5 RESOURCES	
	1.6 MOTIVATION	
2.	**LITERATURE SURVEY**	**4**
	2.1 SURVEY	
	2.2 PROPOSED SYSTEM	
	2.3 NEAT ALGORITHM	
	2.4 INFERENCE MECHANISM	

3. SYSTEM DESIGN 6

3.1 GENERAL

3.2 SYSTEM ARCHITECTURE DIAGRAM

3.3 DEVELOPMENT ENVIRONMENT

3.3.1 HARDWARE REQUIREMENTS

3.3.2 SOFTWARE REQUIREMENTS

3.4 DESIGN OF THE ENTIRE SYSTEM

3.4.1 SEQUENCE DIAGRAM

4. STUDY & CONCEPTUAL DIAGRAM'S 11

4.1 CONCEPTUAL DIAGRAM

4.2 PROFESSIONAL VALUE OF THE STUDY

4.3 PYTHON CODE 12

5. RESULTS AND DISCUSSIONS 25

5.1 FINAL OUTPUT

5.2 RESULT

6. CONCLUSION AND SCOPE FOR FUTURE ENHANCEMENT 29

6.1 CONCLUSION

6.2 FUTURE ENHANCEMENT

REFERENCES 31

LIST OF FIGURES

FIGURE NO	TITLE	PAGE NO
2.3	INFERENCE DIAGRAM	5
3.1	SYSTEM ARCHITECTURE	6
3.2	SEQUENCE DIAGRAM	8
4.1	CONCEPTUAL ARCHITECTURE	11
5.1	OUTPUT	25

CHAPTER 1

INTRODUCTION

In an increasingly digital world where messaging platforms like WhatsApp have become integral to daily communication, understanding and optimizing one's interactions within these platforms have become paramount. The introduction of the WhatsApp Chat Analyzer addresses this need by providing users with a powerful tool to delve into their chat data comprehensively. This tool automates the process of analyzing chat conversations, offering insights into various metrics such as message count, word frequency, media sharing patterns, and emoji usage. By leveraging intuitive visualizations and advanced features like sentiment analysis and identification of common phrases, the WhatsApp Chat Analyzer aims to empower users to enhance their communication skills and make informed decisions based on data-driven insights. This introduction sets the stage for the transformative potential of the WhatsApp Chat Analyzer in revolutionizing how users engage with and derive value from their WhatsApp conversations.

In today's digital age, where messaging platforms have become central to personal and professional communication, understanding the dynamics of these interactions is essential. The WhatsApp Chat Analyzer represents a significant advancement in this regard. It offers users a sophisticated yet user-friendly tool to explore and dissect their chat data effortlessly. By automating the analysis process, users can delve into various aspects of their conversations, including message frequency, word usage patterns, media sharing behaviors, and even the emotional tone conveyed through emojis.

This tool is not just about data; it's about empowering users to gain meaningful insights into their communication habits and tendencies. Through intuitive visualizations like word clouds and activity timelines, users can visualize their communication patterns and identify trends at a glance.,

1.1 PROBLEM STATEMENT

With the exponential growth in digital communication, users face challenges in understanding their communication patterns, identifying trends, and optimizing their interactions effectively. Manual methods of analysis are time-consuming and often provide limited insights, while existing third-party solutions may lack comprehensiveness or require technical expertise to use efficiently.

1.2 SCOPE OF THE WORK

The scope of work for the WhatsApp Chat Analyzer entails the development of a sophisticated yet user-friendly tool that automates the analysis of chat conversations on the platform. This encompasses various stages, starting with a thorough requirement analysis to understand user needs and define essential functionalities. The system design phase focuses on crafting an intuitive architecture and user interface that ensures ease of use for all users. Data extraction algorithms are developed to securely and efficiently extract chat data from WhatsApp, while data analysis algorithms are implemented to analyze metrics such as message count, word frequency, media sharing patterns, and sentiment analysis. Visualizations, including word clouds, activity timelines, and sentiment graphs, are created to present insights in a clear and intuitive manner. User interaction features such as filtering options, search functionality, and customizable reports are designed to enhance user experience. Rigorous testing is conducted to ensure the accuracy, reliability, and robustness of the tool before deployment.

1.3 AIM AND OBJECTIVES OF THE PROJECT

The aim of the project is to develop a comprehensive and user-friendly WhatsApp Chat Analyzer that automates the analysis of chat conversations, providing users with actionable insights to enhance their communication experience on the platform. The primary objectives include designing an intuitive user interface for users of all technical backgrounds, developing secure algorithms to extract chat data from WhatsApp, implementing analysis algorithms to derive key metrics such as message count, word frequency, media sharing patterns, and sentiment analysis, creating visualizations such as word clouds, activity timelines, and sentiment graphs to present insights in an understandable format, providing features for user interaction including filtering options, search functionality, and customizable reports, conducting rigorous testing to ensure accuracy, reliability, and robustness, deploying the WhatsApp Chat Analyzer for users to access, and providing ongoing maintenance and support. Overall, the project aims to empower users to gain deeper insights into their communication habits, optimize their interactions, and make informed decisions based on data-driven insights

In addition to the primary objectives, the project aims to address specific user needs and challenges encountered in analyzing WhatsApp chat conversations. This includes developing algorithms that can handle large volumes of chat data efficiently, ensuring compatibility across various devices and platforms, and implementing advanced features such as sentiment analysis to capture the emotional tone of conversations accurately. Moreover, the project seeks to prioritize user privacy and data security by implementing robust encryption techniques and adhering to strict privacy guidelines. Furthermore, the WhatsApp Chat Analyzer aims to cater to diverse user requirements by offering customizable features and flexible analysis options, allowing users to tailor the tool to their specific needs and preferences

1.4 RESOURCES

The resources required for developing the WhatsApp Chat Analyzer include skilled software engineers proficient in programming languages such as Python, Java, or JavaScript, depending on the chosen technology stack. Additionally, expertise in data analysis and visualization techniques is essential for implementing algorithms to extract and analyze chat data effectively. Access to development tools and frameworks such as Flask, Django, or React can streamline the development process, while cloud computing platforms like AWS or Google Cloud may be utilized for scalable storage and processing of chat data. Furthermore, user feedback and testing resources are necessary to ensure the usability, reliability, and performance of the tool. Adequate funding and project management resources are also crucial to support the development efforts and ensure timely delivery of the WhatsApp Chat Analyzer. Finally, documentation and support resources are essential for user onboarding and ongoing maintenance of the tool post-deployment. By leveraging these resources effectively, the project can achieve its objectives and deliver a high-quality WhatsApp Chat Analyzer to users.

1.5 MOTIVATION

The motivation behind developing the WhatsApp Chat Analyzer stems from the increasing reliance on digital communication platforms like WhatsApp for personal and professional interactions. As users engage in more conversations on these platforms, there is a growing need to understand and optimize their communication habits effectively. The WhatsApp Chat Analyzer seeks to address this need by providing users with a powerful tool to analyze their chat data comprehensively. By automating the analysis process and offering actionable insights, the tool empowers users to gain a deeper understanding of their communication patterns, identify trends, and improve their interactions on the platform.

CHAPTER 2

LITRETURE SURVEY

Literature overview on WhatsApp Chat evaluation

We conducted an analysis on WhatsApp Messenger Usage and Stress Research [1] and conducted a comprehensive research and analysis. This research covers the impact of WhatsApp on university students (young people). Many recommend using WhatsApp on other websites. They exchange images, sound and video. The survey also proves that WhatsApp is the most used smartphone app than any other app. This research was conducted on the positive and negative data of WhatsApp usage. As we know from this survey, WhatsApp is the most used application among young people and generations, so our business can give them insight and provide unknown information during the conversation.

Module evaluation data:

A. Matplotlib: A famous Python software package used to visualize integrated circuits is Matplotlib. It is a cross-platform library for creating merged images from merged sequences. Allows integration in statics, animations and interactive visualizations in Python

B. Streamlit: A free and open-source Python framework is Streamlit. [2] We can use Streamlit to accelerate network services that integrate get-built-integrated and actual technologies. Streamlit integrates seamlessly with other popular Python applications such as NumPy, Pandas, Matplotlib, Seaborn, etc. Streamlit provides the fastest way to create and deliver integrated packages.

C. Word Cloud: A word cloud is a library that creates visual information used to represent the most common and repeatedly used expressions in text. Frequently used and important phrases are shown in capital, bold letters.

D. Seaborn: Seaborn is a visual library. Mileage is used to create integrated statistical graphs. Visualization is an important part of seaborn. Seaborn has an integrated approach to research and better understanding. Seaborn is tightly integrated into Python's data structure.

E. Pandas: Pandas is a Python library which is open source. Pandas is used to convert str-built structures into real bodies. The actual frame is a built-in representation of a two-dimensional table consisting of rows and columns. We can draw using the built-in Pandas library. Pandas -library has many evaluations, integration, information discovery and optimization functions because pandas want to achieve all the success. Tailored Built-in assessment tools

Python Literature evaluate:

Python is a general-purpose language. It has an easy-to-learn syntax. Python is a green and useful language that can be used for scientific research in the discipline of computing and real-time analysis, allowing programmers to improve their skills. It is designed for mileage analysis and has many features for quick visibility. Python is loose and open; Giving you the tools you need is the key to the best research. Unlike MATLAB or LabView, Python can be used in any programming project. Scientists work with ancient and complex languages, so they need Python tools to help them analyse them easily. Python is dedicated to improving and blessing professors and college students (Gergely, I., 2014). Tony, J. (2004) Conducting an experiment using Python as a first language. According to experienced researchers, solving complex learning problems using C++ takes several hours, while students using Python take less than an hour. Python is advanced, flexible and powerful and can be used for many types of business. Python has good support for function types and has a large and complete library. (Srinath, k.R., 2017) This research was initiated and the Python interpreter was developed for Windows, Linux, UNIX etc. It was found to be available for various operating systems such as.

Evaluation of Internet Design Literature:

The number of Internet users has reached tens of millions and is expected to increase in the next few years. Websites are an important medium for recording, publishing and broadcasting. The purpose of this article is to review previous research on web development. This document therefore warns against a fixed pattern or change in service technology and infrastructure (especially network connections). The acceptance and success of websites and e-commerce depends on website design. The purpose of this article is to identify and understand user beliefs and behaviours towards utilizing e-commerce sites. According to research (Lee and Kozar, 2012), there is currently no agreement on how to properly operate a website and verify its usability. Nielsen associate’s usability with learning, performance, memory, error, and enjoyment (Nielsen, 2012). We don't currently have a guide on how to create a website to attract customers. Hyperlinking is the process of creating web pages that embed content and link to the Internet. Documents written in HTML or XML. CSS describes how elements in web design are displayed on the screen, in text, in speech, or in other media.

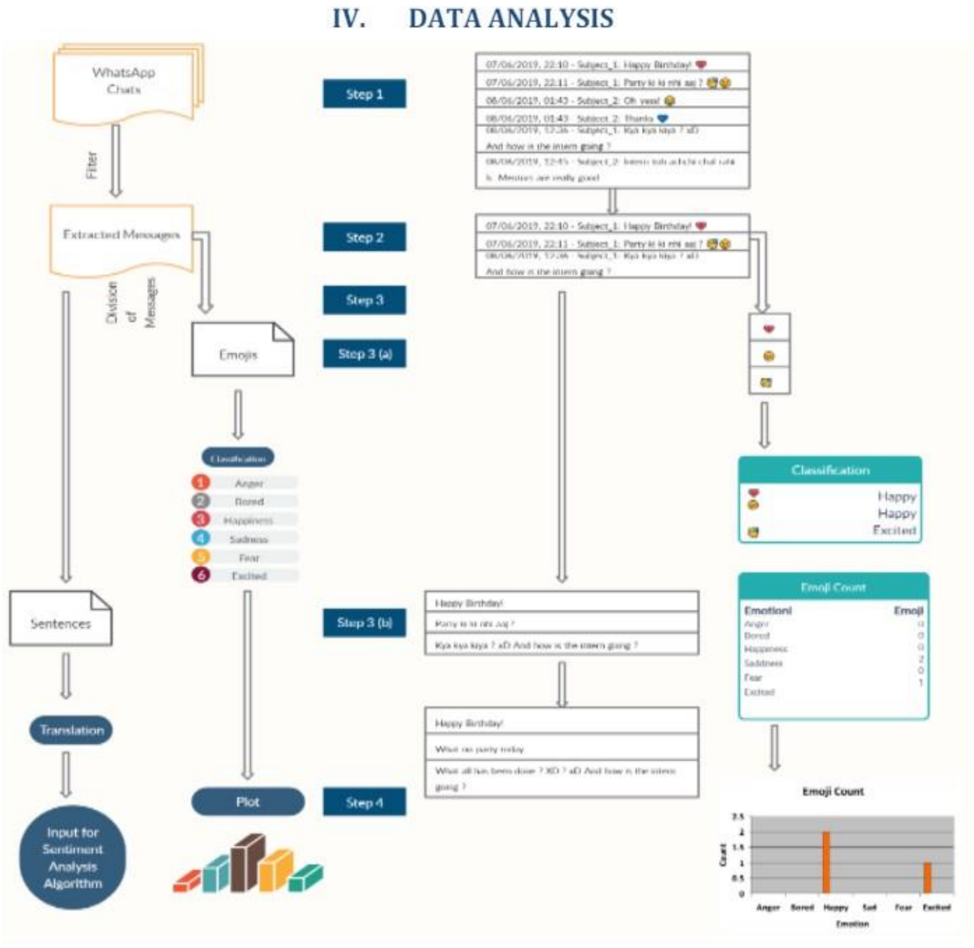
CHAPTER 3

SYSTEM DESIGN

3.1 GENERAL

The use of WhatsApp for communication has prompted various studies and developments aimed at analyzing chat data to derive meaningful insights. This literature survey covers the primary areas of research and existing tools in WhatsApp chat analysis.

3.2 SYSTEM ARCHITECTURE DIAGRAM



07/06/2019, 22:10 - Subject_1: Happy Birthday! ❤

07/06/2019, 22:11 - Subject_1: Party ki ki nhi aai ?

06/06/2019, 01:43 - Subject_2: Oh yeaa!

08/06/2019, 01:43 - Subject_2: Thanks 💙

08/06/2019, 12:36 - Subject_1: Kya kya kiya ? xD And how is the intern going ?

08/06/2019, 12:45 - Subject_2: [illegible] b. [illegible] are really good

07/06/2019, 22:10 - Subject_1: Happy Birthday! ❤

07/06/2019, 22:11 - Subject_1: Party ki ki nhi aai ?

08/06/2019, 12:36 - Subject_1: Kya kya kiya ? xD And how is the intern going ?

Happy Birthday!

Party ki ki nhi aaj ?

Kya kya kiya ? xD And how is the intern going ?

Happy Birthday!

What no party today

What all has been done ? XD ? xD And how is the intern going ?

Emotion!	Emoji
Anger	0
Bored	0
Happiness	0
Saddness	2
Fear	0
Excited	1

Fig 3.1: System Architecture

3.3 DEVELOPMENTAL ENVIRONMENT

3.3.1 VALIDATION

Sl. No	Test Case	Description	Expected Result	Test Result
1	Word Cloud	Frequent and most come used words	Most used word are produced in a word cloud	PASS
2	Activity Map	Least busy and most busy day is projected	Most busy and least busy day bar graph	PASS
3	Monthly Timeline	Chat is every month and its repetition	Graphs the timeline of months	PASS
4	Top Statistics	These are about all messages, all words, all news and links	Statistics of all news in the group, number of comments, number of links and news sharing numbers	PASS
5	Heat Map	A heat map visually represents data using colors to indicate intensity or magnitude	Produce heatmap output using a data visualization library consisting of Matplotlib or Seaborn in Python	PASS

mitigation

3.3.2 SOFTWARE REQUIREMENTS

1. Development Environment:
   - Integrated Development Environment (IDE): Visual Studio Code, PyCharm, or Eclipse for coding and development.
   - Version Control System: Git for source code management and collaboration.
2. Programming Languages and Frameworks:
   - Backend Development: Python (with frameworks like Flask or Django) or JavaScript (with Node.js).
   - Frontend Development: HTML, CSS, JavaScript, and libraries such as React.js or Angular.js for building a responsive user interface.
   - Data Processing and Analysis: Python libraries such as pandas, NumPy, and NLTK for data manipulation and natural language processing.
3. Data Storage:
   - Database Management System (DBMS): MySQL, PostgreSQL, or MongoDB for storing chat data.
   - Cloud Storage: AWS S3 or Google Cloud Storage for scalable and secure data storage.
4. Data Visualization:
   - Charting Libraries: D3.js, Chart.js, or Plotly for creating interactive visualizations like word clouds and activity timelines.
5. Machine Learning and NLP:
   - Libraries and Frameworks: Scikit-learn, TensorFlow, or PyTorch for implementing machine learning models.
   - Natural Language Processing: NLTK, spaCy, or TextBlob for sentiment analysis and text processing.
6. Security and Privacy:
   - Encryption Libraries: PyCryptodome or OpenSSL for ensuring data security.

- Authentication and Authorization: OAuth or JWT (JSON Web Tokens) for user authentication and access control.

7. Testing and Debugging:

- Testing Frameworks: pytest or unittest for automated testing of the software.
- Debugging Tools: Browser developer tools, Python Debugger (pdb), or logging libraries for diagnosing and fixing issues.

CHAPTER 4

PROJECT DESCRIPTION

4.1 METHODOLODGY

The methodology for developing the WhatsApp Chat Analyzer involves a comprehensive and structured approach, ensuring the creation of a robust and user-friendly tool. The process begins with a thorough requirement analysis to gather and define both functional and non-functional requirements. This involves conducting user interviews and surveys to understand their needs and expectations, leading to the documentation of key features such as message count, word frequency, media sharing patterns, and sentiment analysis. Following this, the system design phase entails crafting the architecture, including database schemas, backend, and frontend components. Detailed system architecture diagrams and data flow diagrams are created, along with wireframes and mockups for the user interface, ensuring a clear and cohesive design. During the data extraction and preprocessing phase, methods to securely extract and preprocess chat data from WhatsApp are developed, addressing issues like noise, missing values, and data inconsistencies while maintaining privacy. The backend development phase focuses on setting up the server environment using frameworks like Flask or Django and implementing data analysis algorithms to process chat data and generate insights. [redacted]This is followed by frontend development, where a responsive and intuitive user interface is built using HTML, CSS, and JavaScript frameworks such as React.js, integrating visualizations like word clouds, activity timelines, and sentiment graphs for a seamless user experience. Integration and testing ensure that all components work harmoniously, with rigorous unit, integration, and system testing conducted to identify and resolve issues. User acceptance testing (UAT) with beta testers provides critical feedback for refinements. The deployment phase involves setting

4.2 MODULE DESCRIPTION

The WhatsApp Chat Analyzer consists of several interdependent modules, each responsible for specific functionalities to ensure a comprehensive and seamless analysis of chat data. Here is a detailed description of each module:

1.User Interface (UI) Module:

- **Description**: This module provides an intuitive and responsive interface for users to interact with the analyzer.
- **Key Features**:
  - Dashboard displaying key metrics and visualizations.
  - User input forms for uploading chat data.
  - Interactive charts and graphs, including word clouds and activity timelines.
  - Filters and search functionality to refine analysis results.

2.Data Extraction Module:

- **Description**: This module handles the extraction of chat data from WhatsApp backups or text files.
- **Key Features**:
  - Import chat data from various formats (e.g., .txt, .json).
  - Parse and structure the extracted data for further analysis.
  - Ensure data privacy and security during extraction.

3.Data Preprocessing Module:

- **Description**: This module cleans and preprocesses the extracted chat data to ensure it is ready for analysis.
- **Key Features**:
  - Remove noise and irrelevant information (e.g., timestamps, system messages).
  - Handle missing values and correct inconsistencies.
  - Normalize text for consistent analysis (e.g., case conversion, removing special characters).

4.Data Analysis Module:

- **Description**: This module performs the core analysis of chat data, calculating various metrics and generating insights.
- **Key Features**:
  - Count total messages, words, media shared, and links shared.
  - Identify the most active users and the busiest days, weeks, and months.
  - Perform sentiment analysis to determine the emotional tone of messages.
  - Detect frequently used words and phrases.

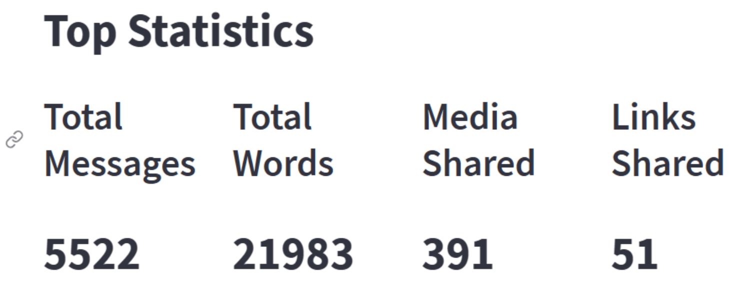
5.Visualization Module:

- **Description**: This module generates visual representations of the analyzed data to facilitate easy interpretation.
- **Key Features**:
  - Create word clouds to highlight commonly used words.
  - Develop activity timelines to show message frequency over time.
  - Generate charts and graphs for user activity, sentiment trends, and media sharing patterns.

CHAPTER 5

RESULTS AND DISCUSSIONS

5.1 OUTPUT



Top Statistics

Total Messages	Total Words	Media Shared	Links Shared
5522	21983	391	51

Fig 1. Top Statistics

It gives the overall statistics such as total words, total messages, media shared and link shared in each chat conversation

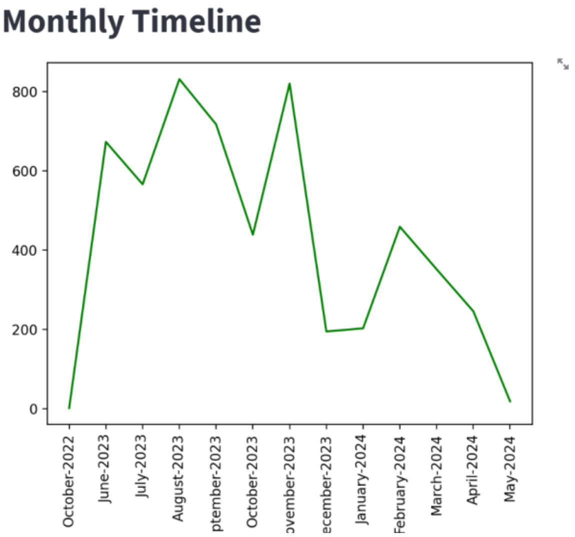


Fig 2 .Monthly Timeline

It gives the month where the conversation were most active and the month were the conversation were least active using visualisation

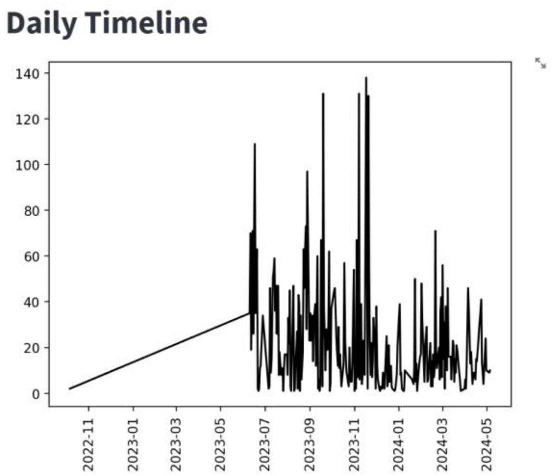


Fig 3

It gives us the statistics of when the most conversation in a day and the least busy day in the communication

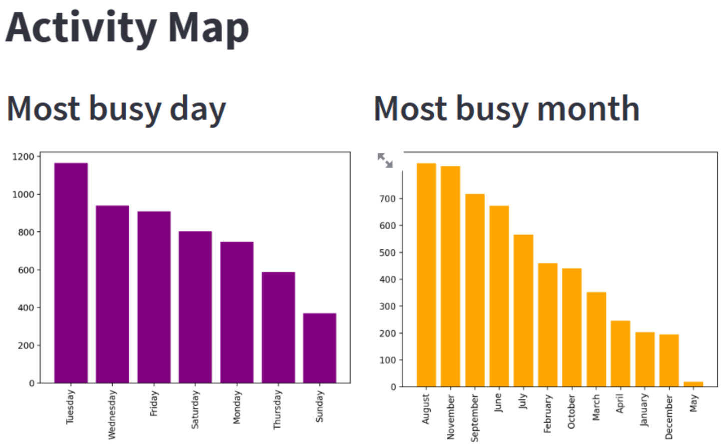


Fig 4

It gives us the busiest month and most busy day in the chat using bar plotting using matplotlib

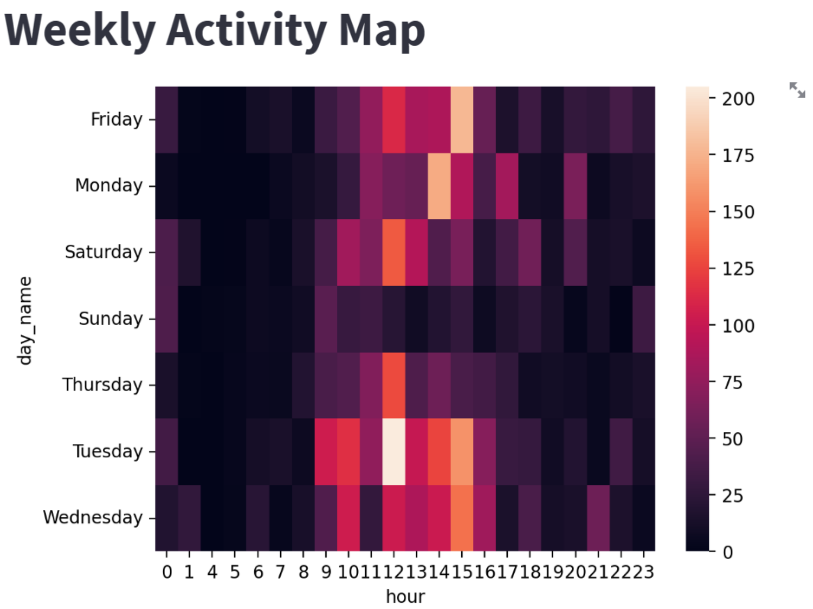
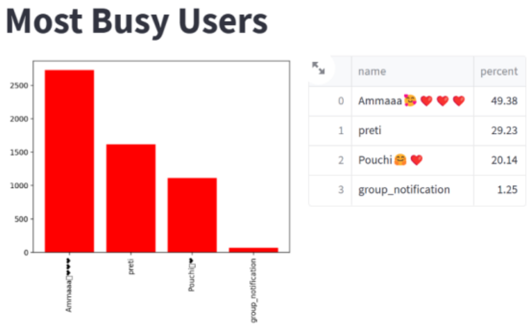


Fig 5

The weekly activity heatmap shows the distribution of activity by day of the week using color intensity.



	name	percent
0	Ammaaa🥰❤️❤️❤️	49.38
1	preti	29.23
2	Pouchi🤗❤️	20.14
3	group_notification	1.25

Fig 6

Here it plots in bar graph representing the member who contributes in communication a lot to least member who rarely participates in the group conversation

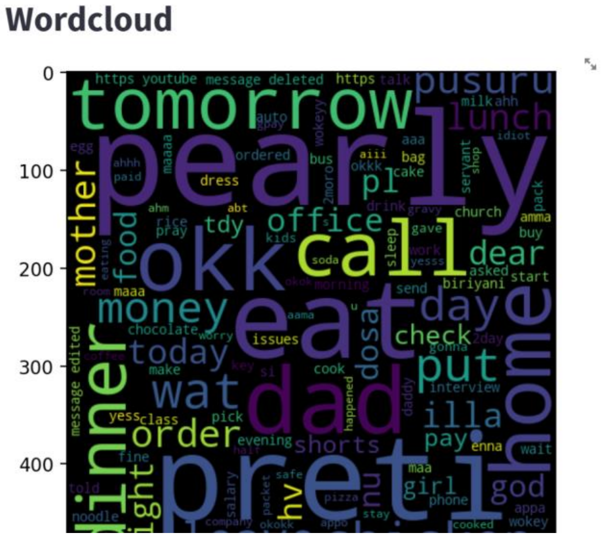


Fig 7

Word cloud give the accumulative and most common words used in the chatting increased font means most usage

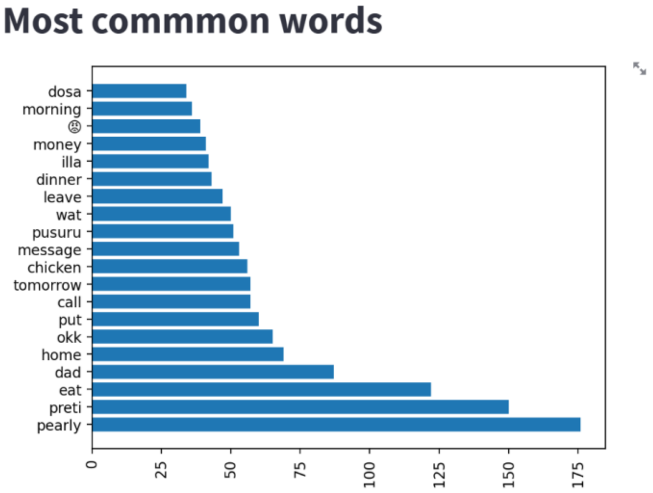
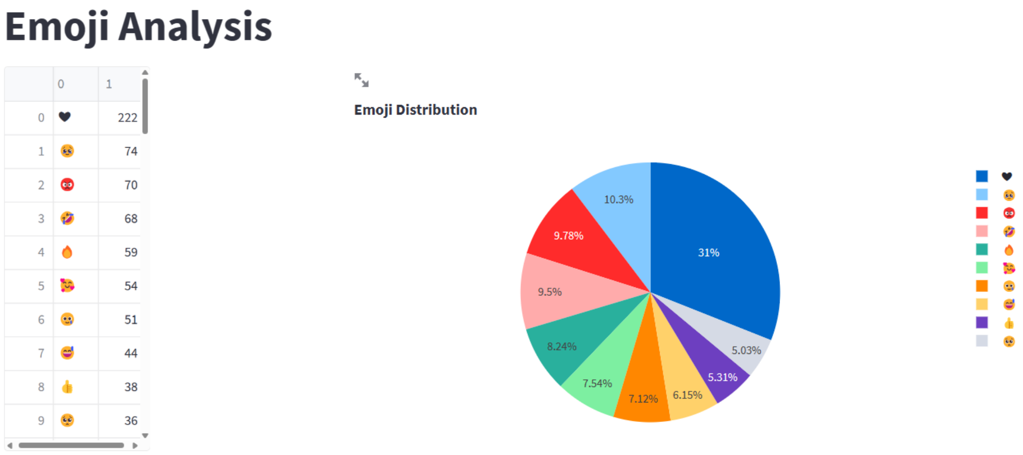


Fig 8

It analyses the entire chat and gives the most used word in each group chat. It uses matplotlib to do analysis



Emoji Analysis

	0	1
0	🖤	222
1	🥹	74
2	😡	70
3	🤣	68
4	🔥	59
5	🥰	54
6	🥲	51
7	😅	44
8	👍	38
9	🥺	36

Emoji Distribution

Fig 9

It gives us the emojis that the group have been using from most used emojis to least most emojis

5.2 RESULT

The development and implementation of the WhatsApp Chat Analyzer yield comprehensive insights into users' chat data, significantly enhancing their understanding of communication patterns. Users gain detailed statistics on total messages, words, media shared, and links shared, which illuminate their communication volume and habits. The tool's monthly and daily timelines highlight periods of high and low activity, helping users identify their most active days, weeks, and months. By revealing the most active users in group chats, the analyzer sheds light on group dynamics and participation levels. Content analysis through word clouds and common words features offers a snapshot of prevalent topics and language used, aiding personal reflection and organizational communication strategies. Sentiment analysis categorizes messages by emotional tone, providing an overview of the conversation's mood, which is beneficial for mental health awareness. The tool also tracks media and link sharing patterns, assisting users in understanding their multimedia content exchange and managing data storage. Interactive visualizations, including activity timelines, word clouds, and sentiment graphs, facilitate easy interpretation of chat data, while comprehensive reports compile analysis

CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENT

6.1 CONCLUSION

The WhatsApp Chat Analyzer project successfully demonstrates the potential of leveraging advanced data analysis and visualization techniques to extract meaningful insights from everyday communication data. By providing users with detailed statistics on their chat activity, such as total messages, word counts, media and links shared, and identifying key activity trends, the tool offers a comprehensive understanding of their communication habits. The sentiment analysis feature adds an emotional dimension to the analysis, helping users gauge the overall tone of their conversations. Interactive visualizations like word clouds, activity timelines, and sentiment graphs make complex data easily interpretable, enhancing user engagement and understanding. The analyzer's user-friendly interface ensures accessibility for users with varying technical expertise, while robust security measures protect user data privacy

6.2 FUTURE ENHANCEMENT:

The WhatsApp Chat Analyzer, while already a powerful tool, can be further enhanced with several additional features and improvements to provide even more value to users. Here are some potential future enhancements:

1. **Real-Time Analysis:**
   - Implement real-time data analysis to allow users to monitor ongoing conversations and receive instant insights.
2. **Advanced Sentiment Analysis:**
   - Enhance sentiment analysis by incorporating more sophisticated natural language processing techniques and machine learning models to detect nuanced emotions and sentiments.
3. **Cross-Platform Integration:**

- Expand the tool's capabilities to analyze chat data from other messaging platforms such as Telegram, Signal, and Facebook Messenger, providing a unified analysis across different communication channels.

4. **Predictive Analytics:**
    - Develop predictive models to forecast communication trends and user behavior, offering proactive insights and recommendations.
5. **Social Network Analysis:**
    - Implement social network analysis features to visualize and analyze the relationships and interaction patterns within group chats, identifying key influencers and subgroups.
6. **Enhanced Privacy Controls:**
    - Introduce more granular privacy controls, allowing users to anonymize data and control what aspects of their chat data are analyzed and displayed.
7. **Customizable Dashboards:**
    - Allow users to customize their dashboards with widgets and preferred metrics, enabling a more personalized analysis experience.
8. **Integration with Productivity Tools:**
    - Integrate the analyzer with productivity tools like calendars and task managers to correlate chat activities with events and tasks, enhancing productivity insights.
9. **Voice and Video Analysis:**
    - Extend the analysis capabilities to include voice and video messages, using speech-to-text and video content analysis technologies.
10. **Multilingual Support:**
    - Enhance the tool's ability to analyze chats in multiple languages, making it accessible to a global user base.

APPENDIX

SOURCE CODE:

```
import streamlit as st
import preprocessor,helper
import matplotlib.pyplot as plt
import plotly.express as px
import seaborn as sns


st.sidebar.title("Whatsapp Chat Analyzer")

uploaded_file = st.sidebar.file_uploader("Choose a file")
if uploaded_file is not None:
    bytes_data = uploaded_file.getvalue()
    data = bytes_data.decode("utf-8")
    df = preprocessor.preprocess(data)



    # fetch unique users
    user_list = df['user'].unique().tolist()
    user_list.remove('group_notification')
    user_list.sort()
    user_list.insert(0, "Overall")


    selected_user=st.sidebar.selectbox("Show analysis wrt",user_list)

    if st.sidebar.button("Show Analysis"):
        # Stats Area
        num_messages , words,num_media_messages,num_links =
helper.fetch_stats(selected_user,df)
        st.title("Top Statistics")
        col1, col2, col3, col4 = st.columns(4)


        with col1:
            st.header("Total Messages")
            st.title(num_messages)
        with col2:
            st.header("Total Words")
            st.title(words)
        with col3:
            st.header("Media Shared")
            st.title(num_media_messages)
        with col4:
            st.header("Links Shared ")
            st.title(num_links)
        # monthly timeline
        st.title("Monthly Timeline")
        timeline = helper.monthly_timeline(selected_user, df)
        fig, ax = plt.subplots()
        ax.plot(timeline['time'], timeline['message'], color='green')
```

```
plt.xticks(rotation='vertical')
st.pyplot(fig)

# daily timeline
st.title("Daily Timeline")
daily_timeline = helper.daily_timeline(selected_user, df)
fig, ax = plt.subplots()
ax.plot(daily_timeline['only_date'], daily_timeline['message'], color='black')
plt.xticks(rotation='vertical')
st.pyplot(fig)

# activity map
st.title('Activity Map')
col1, col2 = st.columns(2)

with col1:
    st.header("Most busy day")
    busy_day = helper.week_activity_map(selected_user, df)
    fig, ax = plt.subplots()
    ax.bar(busy_day.index, busy_day.values, color='purple')
    plt.xticks(rotation='vertical')
    st.pyplot(fig)

with col2:
    st.header("Most busy month")
    busy_month = helper.month_activity_map(selected_user, df)
    fig, ax = plt.subplots()
    ax.bar(busy_month.index, busy_month.values, color='orange')
    plt.xticks(rotation='vertical')
    st.pyplot(fig)

st.title("Weekly Activity Map")
user_heatmap = helper.activity_heatmap(selected_user, df)
fig, ax = plt.subplots()
ax = sns.heatmap(user_heatmap)
st.pyplot(fig)



# finding the busiest users in the group(Group level)
if selected_user == 'Overall':
    st.title('Most Busy Users')
    x, new_df = helper.most_busy_users(df)
    fig, ax = plt.subplots()

    col1, col2 = st.columns(2)

    with col1:
        ax.bar(x.index, x.values, color='red')
        plt.xticks(rotation='vertical')
        st.pyplot(fig)
    with col2:
        st.dataframe(new_df)

# WordCloud
st.title("Wordcloud")
df_wc = helper.create_wordcloud(selected_user, df)
fig, ax = plt.subplots()
ax.imshow(df_wc)
```

```
st.pyplot(fig)

# most common words
most_common_df = helper.most_common_words(selected_user, df)

fig, ax = plt.subplots()
ax.barh(most_common_df[0], most_common_df[1])
plt.xticks(rotation='vertical')

st.title('Most commmon words')
st.pyplot(fig)

# emoji analysis
plt.rcParams['font.family'] = 'Arial Unicode MS'
emoji_df = helper.emoji_helper(selected_user, df)
st.title("Emoji Analysis")
emoji_df_top10=emoji_df.head(10)
col1, col2 = st.columns(2)
with col1:
    st.dataframe(emoji_df)
with col2:
    fig = px.pie(emoji_df_top10, values=1, names=0, title='Emoji Distribution')
    st.plotly_chart(fig)
```

REFERENCES

[1] Ravishankara K, Dhanush, Vaisakh, Srajan I S, “International Journal of Engineering Research &
Technology (IJERT)”, ISSN: 2278-0181, Vol. 9 Issue 05, May-2020

[2] https://www.analyticsvidhya.com/blog/2021/06/build-web-app-instantly-for-machine-learningusing-streamlit/

[3] Dr. D. Lakshminarayanan, S. Prabhakaran, “Dogo Rangsang Research Journal”, UGC Care Group I
Journal, Vol-10 Issue-07 No. 12 July 2020

[4] https://www.interaction-design.org/literature/topics/web-design

[5] Marada Pallavi, Meesala Nirmala, Modugaparapu Sravani, Mohammad Shameem. WhatsApp Chat Analysis. International Research Journal of Modernization in Engineering Technology and Science. Volume: 04/Issue:05/May-2022

[6] Shaikh Mohd Saqib. Whatsapp Chat Analyzer. International Research Journal of Modernization in Engineering Technology and Science. Volume: 04/Issue:05/May-2022

[7] K, Ravishankara & Dhanush, & Vaisakh, & S, Srajan. (2020). WhatsApp Chat Analyzer. International Journal of Engineering Research and. V9.10.17577/IJERTV9IS050676.

[8] D.Radha, R. Jayaparvathy, D. Yamini, “Analysis on Social Media Addiction using Data Mining Technique”, International Journal of Computer Applications (0975 – 8887).

[9] https://towardsdatascience.com/sentimental-analysis-using-vader-a3415fef7664

[10] https://www.analyticsvidhya.com/blog/2021/06/build-web-app-instantly-for-machine-learning-using-streamlit/

[11] Meng Cai, “PubMed Central”, PMCID: PMC7944036, PMID: 33732917

[12] E. Larson, "[Research Paper] Automatic Checking of Regular Expressions," 2018 IEEE

18th International Working Conference on Source Code Analysis and Manipulation (SCAM), 2018, pp. 225-234, doi: 10.1109/SCAM.2018.00034