IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Impact Factor: 7.301

Volume 3, Issue 12, May 2023

Unveiling WhatsApp Chat Insights: A Comprehensive Analysis of Conversational Patterns and User Behavior in Streamlit Framework using Machine Learning

Dr. S. P. Pawar, Vaishnavi V. Jathar, Mayeshwari S. More, Lajina S. Devale, Sakshi G. Bhosale, Shaina I. Shaikh

Department of Computer Science and Engineering

SVERI'S College of Engineering, Pandharpur, India

Abstract*: Massive libraries of textual discussions have been created as a result of the widespread usage of instant messaging services like WhatsApp, which has revolutionized how people communicate. This study goes into the world of WhatsApp chat insights, offering a thorough examination of user behavior and linguistic trends. We seek to uncover important insights buried in WhatsApp chat data using data analytics approaches, natural language processing (NLP) algorithms, and machine learning. In order to identify patterns, trends, and user behavioral traits from WhatsApp chat records, this study proposes a methodical technique.*

Keywords: User behavior, Natural language processing (NLP), Machine learning, Streamlit framework, Statistics

I. INTRODUCTION

1.1 Background and Motivation:

Instant messaging systems have altered the way individuals communicate and connect in today's digital landscape. With its extensive popularity and user base, WhatsApp has established itself as a prominent player in the field of instant messaging. The massive number of WhatsApp conversations is a rich source of data that can provide important insights into human communication patterns and user behavior. Understanding these patterns and behaviors can have far-reaching consequences in fields as diverse as psychology, sociology, marketing, and data science.

This research study was inspired by the ambition to use machine learning and data analytics to unearth hidden knowledge within WhatsApp chat logs. We hope to investigate the underlying structure and dynamics of WhatsApp chats using advanced algorithms and approaches, providing a better understanding of how users engage and interact with this popular messaging network.

1.2 Research Objectives:

The major goal of this research project is to extract relevant insights from WhatsApp chat data using machine learning techniques and Streamlit. The following are the specific research objectives:

Conversational patterns are analysed and visualised in order to find trends in message volume and communication frequency in WhatsApp chats.

Understanding user behavior: We look for behavioral traits including reaction times, degrees of engagement, and discussion participation patterns.

We want to analyze the sentiment and emotional expressions hidden in WhatsApp conversations by using natural language processing methods. Sentiment analysis and emotion identification.

1.3 Research Questions:

To achieve our research objectives, we will address the following key research questions:

DOI: 10.48175/IJARSCT-10670

What are the most common conversational dynamics and patterns in WhatsApp chats?
What are users' reaction times, levels of engagement, and involvement patterns?

1.4 Methodology Overview:

We will use a methodical strategy for data collecting, preparation, and analysis in this study. In order to protect the privacy and identity of the users engaged, WhatsApp chat records will be acquired. After that, preprocessing operations including cleaning, formatting, and anonymization will be performed on the acquired data. To glean insights from the processed conversation data, machine learning algorithms and data analytics methods like NLP, sentiment analysis, topic modelling, and network analysis will be used. An interactive web application for visualizing and presenting the acquired results will be created using the Streamlit framework.

II METHODOLOGY:

2.1 Data Collection:

To gather the necessary data for this research, WhatsApp chat conversations were collected through the platform's built-in export feature. This feature allows users to export their chat history as a file, facilitating the collection of conversation data in a structured format. By utilizing this export functionality, the complete chat content, including text messages, emojis, media files, and attachments, was captured and saved in a designated file format.

The message content, participant data, timestamps, and, in certain circumstances, other chat-specific metadata are all included in the exported file, which is crucial for analysis. This technique of gathering data offers a thorough snapshot of the conversation history, facilitating further analysis and study.

WhatsApp chat chats were exported from the platform in order to acquire the information required for this study. Users can export their chat history as a file with this function, which makes it easier to gather conversation data in a structured way. The whole conversation information, including text messages, emoticons, media files, and attachments, was recorded and saved in a selected file format by using this export functionality.

This study attempts to obtain insights into the dynamics of WhatsApp chats and draw insightful conclusions from the gathered data by using the exported chat data as a useful resource.

2.2 Data Preprocessing:

Regular Expression in Data Preprocessing: A search pattern is produced by a regular expression, which is a collection of characters. They provide for the effective cleaning, formatting, and extraction of data by allowing researchers to establish precise patterns or rules for text manipulation. Regular expressions are especially helpful in handling noisy or unstructured data because they offer a versatile and concise technique to recognize and convert textual patterns.

Text cleaning: Regular expressions assist in removing unnecessary characters from chat data, such as emojis, special symbols, and punctuation. They allow researchers to swap out or get rid of particular patterns that can obstruct further investigation.

Tokenization: Regular expressions can help tokenize chat data by separating it into individual words or "tokens." Researchers can tokenize the text and produce a structured representation of the chat data by establishing the right patterns, which will facilitate additional analysis or the use of natural language processing methods.

Pattern Matching and Extraction: Regular expressions enable researchers to find particular patterns within the conversation data and extract pertinent information. Pattern matching and extraction. For instance, researchers can extract URLs, email addresses, or phone numbers mentioned in the discussions using regular expressions.

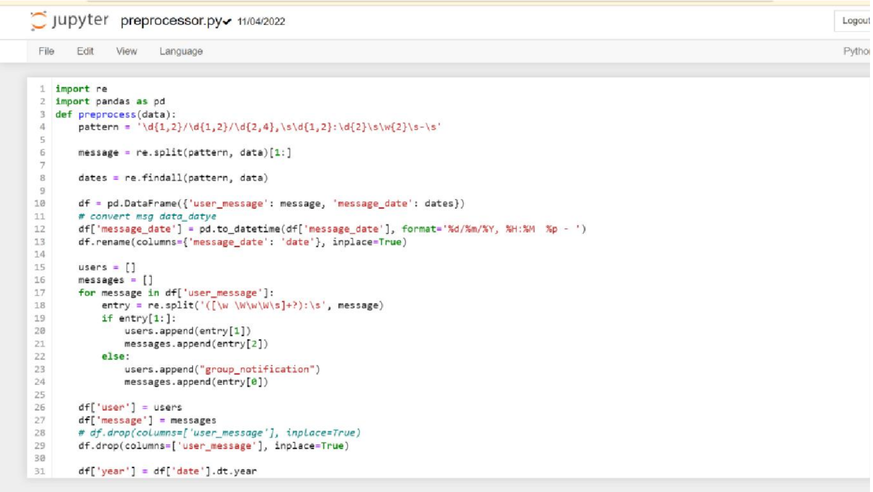
Data Transformation: Regular expressions offer a way to translate conversation data into the forms that are needed. They can be used by researchers to standardise date and time formats in the conversation logs or convert text to lowercase.

III. SYSTEM DEPLOYMENT

Streamlit framework: A Python framework called Streamlit is made for creating dynamic web apps for data processing and visualization. It makes it easier to create user-friendly dashboards and reports without needing to have a lot of web programming knowledge. Developers can swiftly prototype and iterate with Streamlit while easily integrating data

DOI: 10.48175/IJARSCT-10670

science libraries. The framework is a potent tool for exhibiting and disseminating data-driven insights since it allows for real-time updates, simple connection with well-known data science tools, and straightforward deployment.

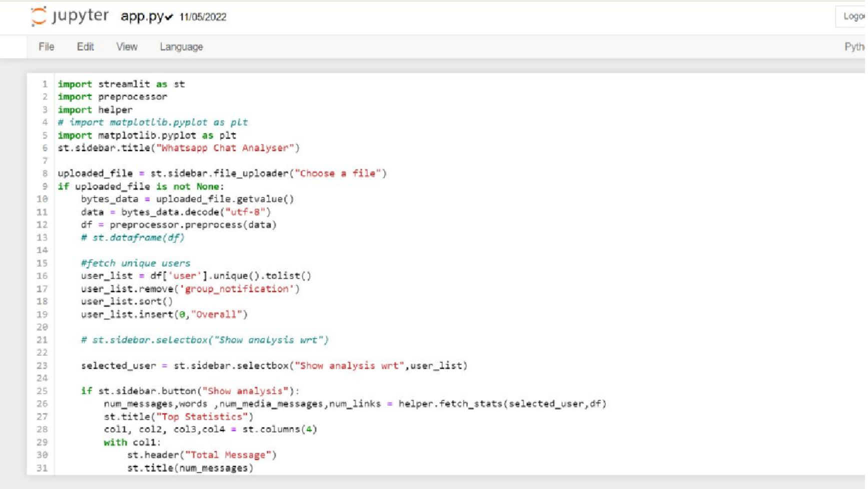


```python
import re
import pandas as pd
def preprocess(data):
    pattern = '\d{1,2}/\d{1,2}/\d{2,4},\s\d{1,2}:\d{2}\s\w{2}\s-\s'

    message = re.split(pattern, data)[1:]

    dates = re.findall(pattern, data)

    df = pd.DataFrame({'user_message': message, 'message_date': dates})
    # convert msg data_datye
    df['message_date'] = pd.to_datetime(df['message_date'], format='%d/%m/%Y, %H:%M  %p - ')
    df.rename(columns={'message_date': 'date'}, inplace=True)

    users = []
    messages = []
    for message in df['user_message']:
        entry = re.split('([\w \W\w\W\s]+?):\s', message)
        if entry[1:]:
            users.append(entry[1])
            messages.append(entry[2])
        else:
            users.append("group_notification")
            messages.append(entry[0])

    df['user'] = users
    df['message'] = messages
    # df.drop(columns=['user_message'], inplace=True)
    df.drop(columns=['user_message'], inplace=True)

    df['year'] = df['date'].dt.year
```

Dataframes in pandas: Similar to a table in a spreadsheet or database, a DataFrame in Python is a two-dimensional, tabular data structure that arranges data into rows and columns. The pandas package, which is widely used for data manipulation and analysis, provides it as a core data structure.

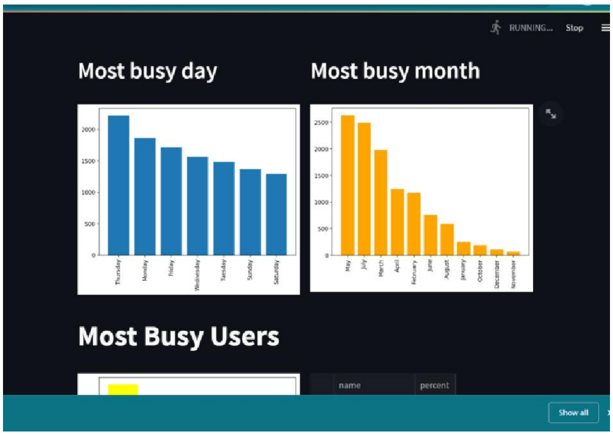
Filtering, sorting, merging, and aggregating are just a few of the operations for manipulating data that are available in DataFrames. Additionally, they support a number of data types, such as CSV, Excel, SQL databases, and others. DataFrames are a well-liked option for data analysis jobs in Python because they offer a clear and effective approach to work with structured data.

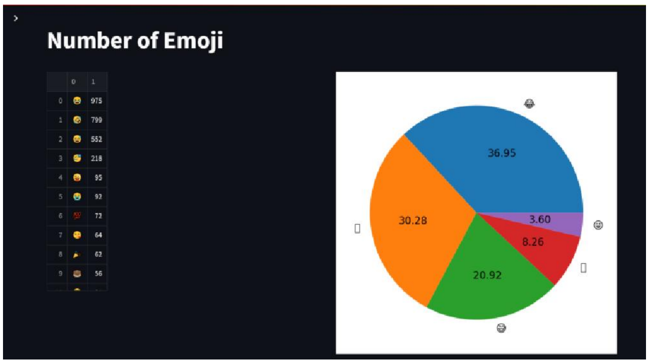
Pycharm: An integrated development environment (IDE) created especially for Python programming is called PyCharm. It is a popular option for Python developers and data scientists since it offers a wide range of tools and capabilities that improve the development process.



```python
import streamlit as st
import preprocessor
import helper
# import matplotlib.pyplot as plt
import matplotlib.pyplot as plt
st.sidebar.title("Whatsapp Chat Analyser")

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)
    # st.dataframe(df)

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

    # st.sidebar.selectbox("Show analysis wrt")

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

    if st.sidebar.button("Show analysis"):
        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 Message")
            st.title(num_messages)
```

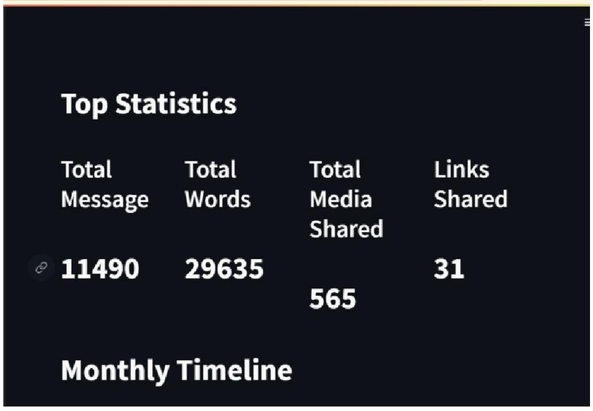
DOI: 10.48175/IJARSCT-10670





Number of Emoji

	0	1
0	😂	975
1	🤣	799
2	😩	552
3	🥱	218
4	😋	95
5	😭	92
6	💯	72
7	🥺	64
8	🎉	62
9	🙈	56



Top Statistics

Total Message	Total Words	Total Media Shared	Links Shared
11490	29635	565	31

Monthly Timeline

IV CONCLUSION:

In this study, the Streamlit framework and machine learning methods were employed to analyze WhatsApp chat data. The project attempted to analyze user behavior, get insights from chat chats, and extract useful information from the data.

An interactive and user-friendly web application was created by using the Streamlit framework to carry out the job. Through this application, users may easily and conveniently study and visualise chat insights. The extraction of significant patterns and trends from the conversation data was made easier by the inclusion of machine learning algorithms.

The raw chat data was cleaned up and transformed using regular expressions, which made it possible to conduct a more precise analysis.

For data processing and analysis, the Python pandas library's DataFrame data format offered a potent tool. Deeper insights and exploration were made possible by its fast handling, filtering, and aggregation of the conversation data.

DOI: 10.48175/IJARSCT-10670

The development process was accelerated by PyCharm's extensive range of capabilities and tools, which was the platform of choice. A simplified and effective workflow was made possible by its code editor, debugging tools, and interface with version control systems.

Overall, this study has shown the value of using regular expressions, the Python ecosystem, the Streamlit framework, and machine learning approaches to analyse WhatsApp chat data. Combining these technologies and approaches made it possible to glean insightful information from chat conversations, including patterns, user behaviour, and other priceless data.

Finally, this research project has demonstrated the possibility of merging machine learning, the Streamlit framework, and other Python tools for efficient data exploration and visualization. It has given useful insights into WhatsApp chat analysis.

V REFERENCES

[1]. Machine Learning For Absolute Beginners by *Oliver Theobal, John Myles White*,Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow by Geron Aurelien,Deep Learning by *Ian Goodfellow, Yoshua Bengio, and Aaron Courville.*

[2]. International Journal of Engineering Research & Technology (IJERT) http://www.ijert.org ISSN: 2278-0181 IJERTV9IS050676 (This work is licensed under a Creative Commons Attribution 4.0 International License.) Published by: www.ijert.org Vol. 9 Issue 05, May-2020, International Research Journal of Modernization in Engineering Technology and Science (Peer-Reviewed, Open Access, Fully Refereed International Journal) Volume:04/Issue:05/May-2022 Impact Factor- 6.752 www.irjmets.com.

[3]. https://www.geeksforgeeks.org/machine-learning/#dp,https://www.geeksforgeeks.org/machine-learning/#nlp, https://towardsdatascience.com,https://www.kaggle.com,https://machinelearningmastery.com

DOI: 10.48175/IJARSCT-10670