WHATSAPP SENTIMENT ANALYSIS

FINAL PROJECT NLP

NATURAL LANGUAGE PROCESSING SPRING 2022

Med Hedi Ben Atig

Julián Oliveros Forero

10 of July de 2022

Table content

1. Introduction

The goal of the project is to detect the sentiment in WhatsApp conversations there will be 3 different categories of sentiment which are positive, negative, and neutral. The program will receive a .txt that contains the messages in a WhatsApp conversation that could be between two people or a group and will give the percentage of the sentiment of each message. After knowing the percentage of the sentiment of each message we will display some graphs that show relevant information to be able it understands the data such as the total number of messages, the global sentiment in all the conversations, the average sentiment per each person in the conversation and the timeline of sentiment during each conversation.

2. Construction of the creation of code

Once we decided to do a project related to NLP for WhatsApp conversations, we started looking for information on google about how we could be able to create an algorithm that solves this problem. After having these analyses, we decided to use the programming language python since we had been working with it all semester and because we find one python library called NLTK According ("Natural Language Toolkit," 2022) "The Natural Language Toolkit, or more commonly NLTK, is a suite of libraries and programs for symbolic and statistical natural language processing (NLP) for English written in the Python programming language". So, this library will be useful to complete our goals. As well we will use streamlit to deploy a web interface and plotly to show interesting graphs about the results.

Once we investigated and defined the technology stack that we will be using for the project we start to create the code which is mainly divided into 3 parts which are data extraction, sentiment prediction, and sentiment interpretation.

The first section of data extraction consists of reading a .txt file that contains all the WhatsApp conversations, each row of this text is a message sent in the following format:

[8:33 pm, 06/07/2022] Julian: <message>

Then we must extract each element of the row and save the different information in a pandas data frame so that we can use it for later steps. In this part, we faced a problem, and sometimes when messages are too large or are sent in the same minute, they will be stored without the initial brackets like this:

[9:35 pm, 06/07/2022] Mohamad: Wanna party?
Sounds weird going out from me I know😊

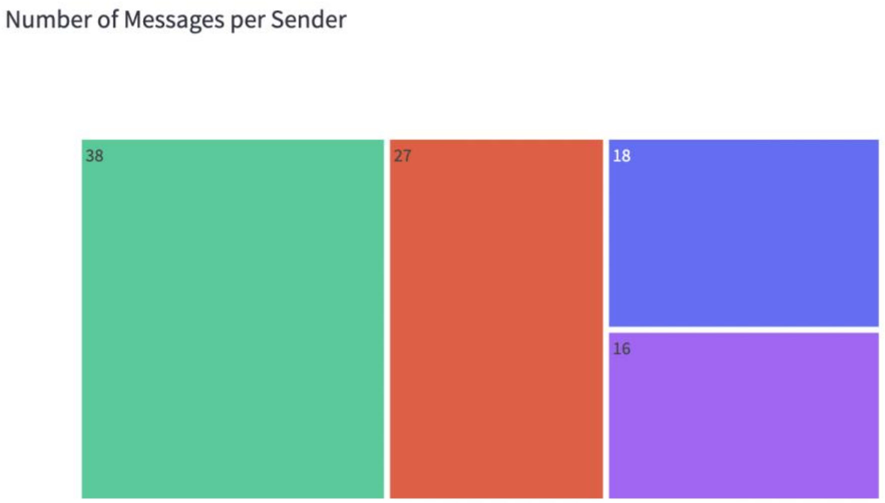
[9:35 pm, 06/07/2022] Aida: what 😹😹😹

So, we need to work on how to deal with this case and what we did was append this line to the previous message since it will have the same date-time attributes. Another problem we had was that we need to take each line and decode it since they are emojis, so it was decoded using “utf-8”. Also, we took some time trying to extract each part of the message and the sender of the message since it is separated by two dots symbol but if there's another symbol like this will cause a problem while doing the split, so this is with we did the split considering only the first double dot. And finally, we were able to store all the incoming information in the Pandas data frame that has the following columns *columns= ['Date & Time', 'Sender', 'Message]'*.

The next section of the code is the sentiment prediction this section consists of implementing the natural language processing to know the sentiment of a message of a string. We look a lot of different libraries that could help us implement this and finally, we found a module of library NLTK name SentimentIntensityAnalyzer that can be found in the GitHub repository on the following link https://github.com/nltk/nltk/blob/develop/nltk/sentiment/sentiment_analyzer.py. This module will help us know the sentiment of a message and it will give us an output of the percentage of each sentiment of the message between positive, neutral, and negative.

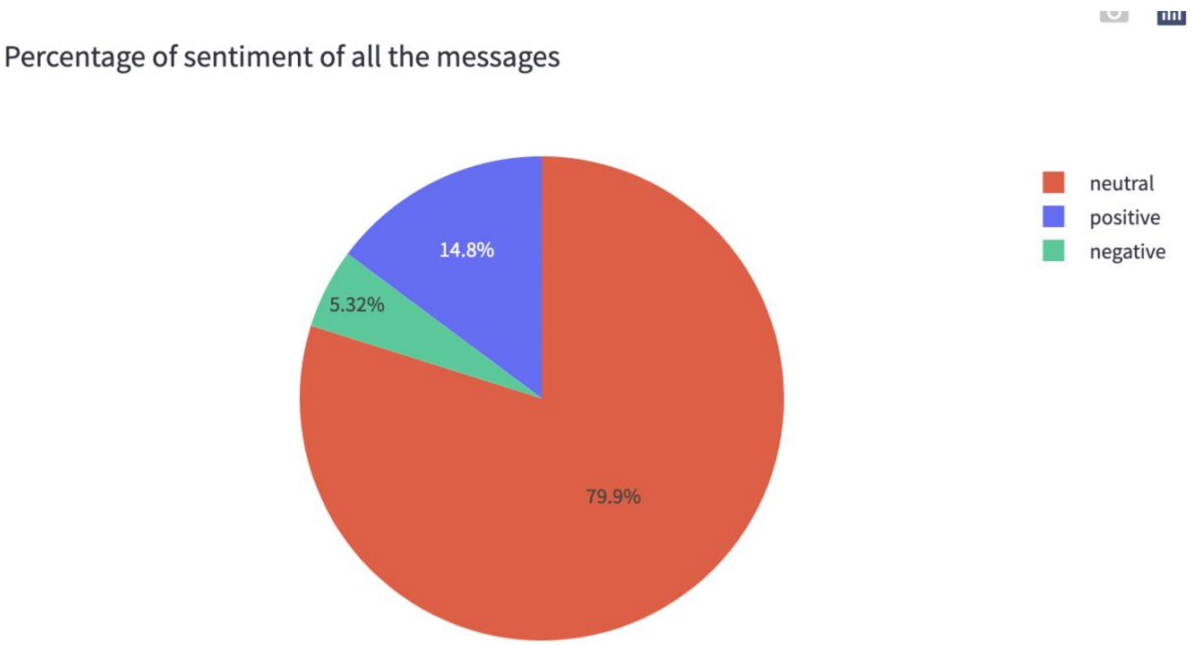
In the final section, we have sentiment interpretation this part consists of taking the results of the algorithm and making graphs and visualizations that will help us understand the output of the algorithm. To make this we create a data frame that will have the following columns *columns= ['Date & Time', 'Sender', 'Message,' positive',' negative',' neutral']*. The new 3 columns will contain the % that corresponds to the message to each category meaning that for one row the sum of these 3 columns is 100% or 1. To analyze we decided to show 4 graphs.

1) A tree map that shows how many messages have been sent to each person involved in the conversation.



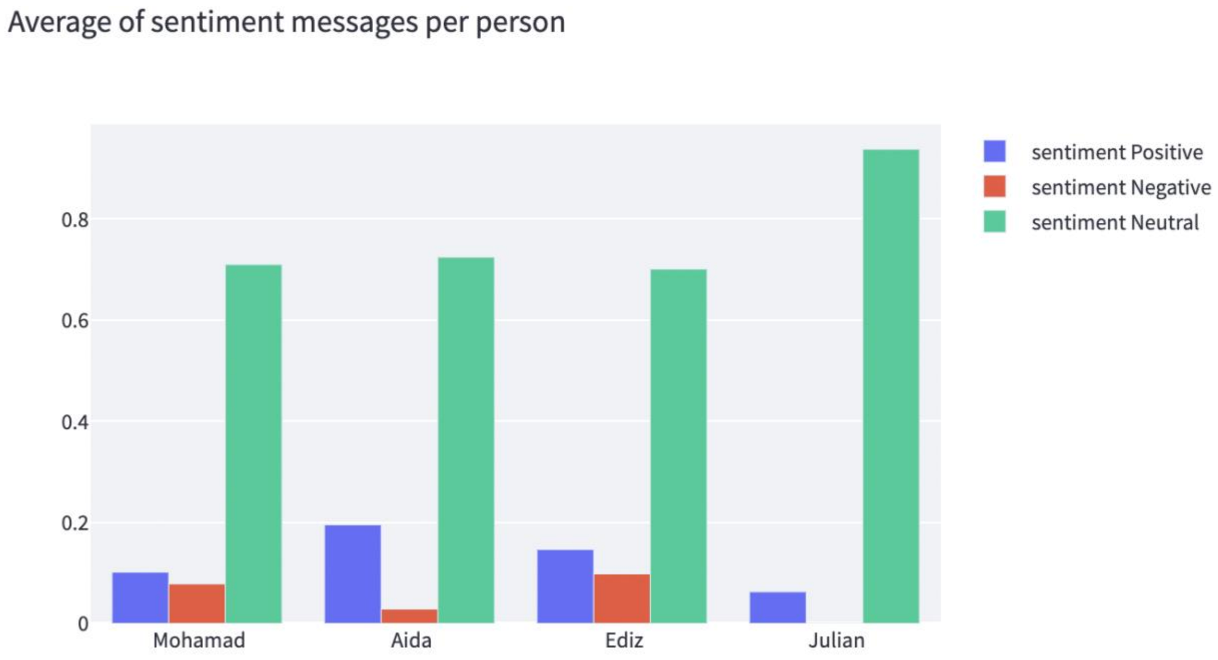
Number of Messages per Sender

2) The second visualization is a pie chart that shows the global percentage of the sentiment of the whole conversation.

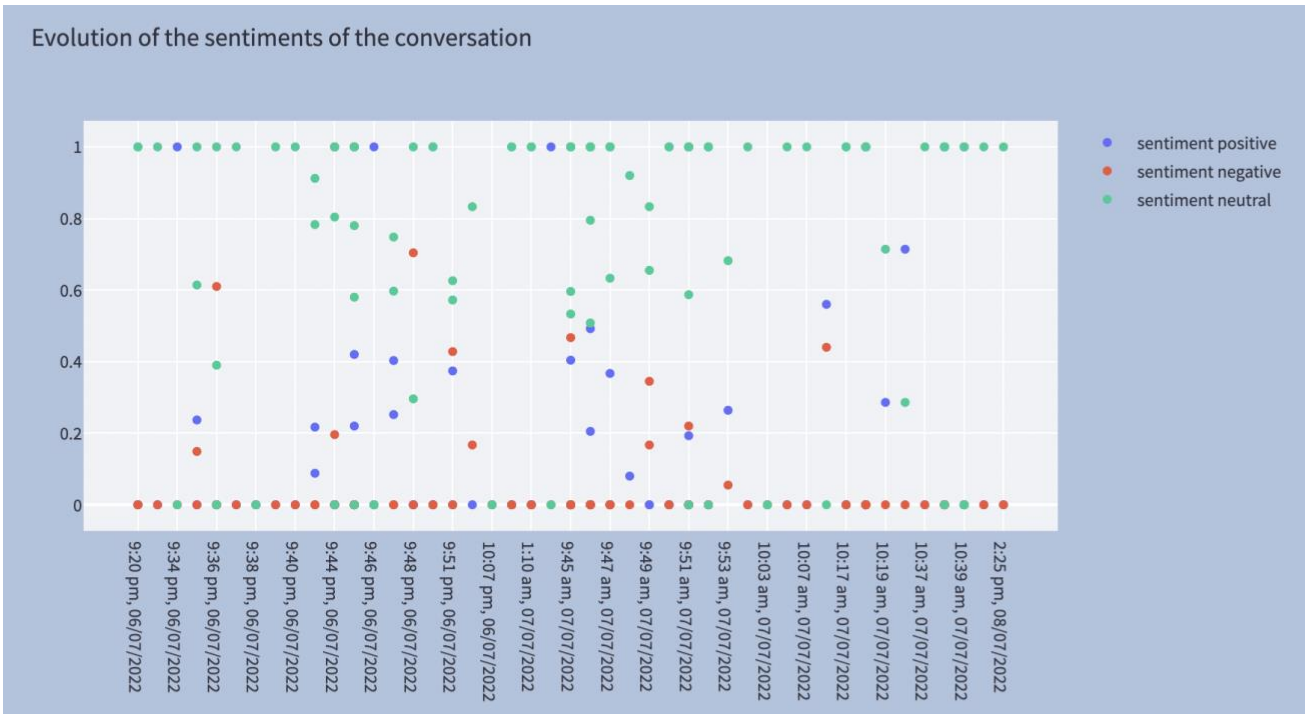


Percentage of sentiment of all the messages

3) The third visualization shows the average of each category of sentiment for each person involved in the conversation.



4) The last graphs show the evolution of the sentiment during all the time of the conversation.



In conclusion, this was a project need a lot of research and investigation to be able to be implemented and how data is extracted is very important and one of the ones we should be more precise about since there could be a lot of different cases that can make data come in the wrong format.

3. Link GitHub Repository

https://github.com/HediAtig/NLP

4. Video Demo

https://www.youtube.com/watch?v=k3UY9nv1wQQ

5. Bibliography

Anwar, A. (2021, January 19). *AnalyzeTheChat—Python-based WhatsApp chat analyzer*. Medium. https://towardsdatascience.com/analyzethechat-python-based-whatsapp-chat-analyzer-cb53951c7ebb

Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit* [Python]. O'Reilly Media, Inc. https://github.com/nltk/nltk/blob/86b11fb75b1e7a7cf60e5bf9140053cd742e6df1/nltk/sentiment/sentiment_analyzer.py (Original work published 2009)

Kharwal, A. (2020, July 16). *NLP For WhatsApp Chats*. Data Science | Machine Learning | Python | C++ | Coding | Programming | JavaScript. https://thecleverprogrammer.com/2020/07/16/nlp-for-whatsapp-chats/

Kharwal, A. (2021, April 9). *WhatsApp Chat Analysis with Python*. Data Science | Machine Learning | Python | C++ | Coding | Programming | JavaScript.

https://thecleverprogrammer.com/2021/04/09/whatsapp-chat-analysis-with-python/

MONDAL, R. (2022). *Machine LeaningLearning projects for beginners* [Jupyter Notebook]. https://github.com/rajkrishna92/Machine-Leaning-projects-for-beginners/blob/6a232919bb5aed016ded83405aa4e51b7403dbcc/WhatsApp%20Chat%20Sentiment%20Analysis.ipynb (Original work published 2021)

Natural Language Toolkit. (2022). In *Wikipedia*. https://en.wikipedia.org/w/index.php?title=Natural_Language_Toolkit&oldid=1080767567

NLTK :: Sample usage for the sentiment. (n.d.). Retrieved July 9, 2022, from https://www.nltk.org/howto/sentiment.html

WhatsApp Analyzer | WhatsApp Group Chat Analysis using Python. (2021, April 30). *Analytics Vidhya*. https://www.analyticsvidhya.com/blog/2021/04/whatsapp-group-chat-analyzer-using-python/