

main

Machine-Learning-Projects-with-Python-for-Beginners / WhatsApp Chat Sentiment Analysis using Python / WhatsApp_Chat_Sentiment_Analysis.ipynb

Go to file

...








Rasel1435 WhatsApp Chat Sentiment Analysis

Latest commit 3038ac6 43 seconds ago [History](#)

1 contributor

658 lines (658 sloc) | 20.9 KB

  Raw Blame   

WhatsApp Chat Sentiment Analysis using Python

WhatsApp is a great source of data to analyze many patterns and relationships between two or more people chatting personally or even in groups. If you want to know how we can analyze the sentiments of a WhatsApp chat, this article is for you. In this article, I will walk you through the task of WhatsApp chat sentiment analysis using Python.

WhatsApp Chat Sentiment Analysis

To analyze the sentiments of a WhatsApp chat, we need to collect data from WhatsApp. Most of you must be using this messaging app, so to collect data about your chat, simply follow the steps mentioned below

In [44]:

```
# For iPhone:
# 1.Open your chat with a person or a group
# 2.Just tap on the profile of the person or the group
# 3.You will see an option to export chat down below
# For Android:
# 1.Open your chat with a person or a group
# 2.Click on the three dots above
# 3.Click on more
# 4.Click on the export chat
```

You will see an option to attach media while exporting your chat. For simplicity, it is best not to attach media. Finally, enter your email and you will find your WhatsApp chat in your inbox.

WhatsApp Chat Sentiment Analysis using Python

Now let's start with the task of WhatsApp chat sentiment analysis with Python. I'll start this task by defining some helper functions because the data we get from WhatsApp is not a dataset that is ready to be used for any kind of data science task. So, to prepare your data for the sentiment analysis task, just define all the functions as defined below

Import library

In [45]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import re
import emoji
import nltk

from collections import Counter
from PIL import Image
from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator

%matplotlib inline
```

Extract Time

In [46]:

```
def date_time(s):
    pattern = '^([0-9]+)(\V)([0-9]+)(\V)([0-9]+), ([0-9]+):([0-9]+)[ ]?(AM|PM|am|pm)? -'
    result = re.match(pattern, s)
    if result:
        return True
    return False
```

Find Authors or Contacts

In [47]:

```
def find_author(s):
    s = s.split(":")
    if len(s) == 2:
        return True
    else:
        return False
```

Finding Messages

```
In [48]: def getDatapoint(line):
splitline = line.split(' - ')
dateTime = splitline[0]
date, time = dateTime.split(", ")
message = " ".join(splitline[1:])
if find_author(message):
    splitmessage = message.split(": ")
    author = splitmessage[0]
    message = " ".join(splitmessage[1:])
else:
    author = None
return date, time, author, message
```

It doesn't matter if you are using a group chat dataset or your conversation with one person. All the functions defined above will prepare your data for the task of sentiment analysis as well as for any data science task. Now here is how we can prepare the data we collected from WhatsApp by using the above functions

```
In [49]: data = []
conversation = 'data\WhatsApp Chat with HRV MLDS Community.txt'
with open(conversation, encoding="utf-8") as fp:
    fp.readline()
    messageBuffer = []
    date, time, author = None, None, None
    while True:
        line = fp.readline()
        if not line:
            break
        line = line.strip()
        if date_time(line):
            if len(messageBuffer) > 0:
                data.append([date, time, author, ' '.join(messageBuffer)])
                messageBuffer.clear()
            date, time, author, message = getDatapoint(line)
            messageBuffer.append(message)
        else:
            messageBuffer.append(line)
```

Now here is how we can analyze the sentiments of WhatsApp chat using Python

```
In [50]: from nltk.sentiment.vader import SentimentIntensityAnalyzer
df = pd.DataFrame(data, columns=["Date", "Time", "Author", "Message"])
df['Date'] = pd.to_datetime(df['Date'])
```

```
data = df.dropna()
sentiments = SentimentIntensityAnalyzer()
data["Positive"] = [sentiments.polarity_scores(
    i["pos"] for i in data["Message"])
data["Negative"] = [sentiments.polarity_scores(
    i["neg"] for i in data["Message"])
data["Neutral"] = [sentiments.polarity_scores(
    i["neu"] for i in data["Message"])
print(data.head())
```

```
   Date      Time      Author \
2 2022-06-07 11:10 PM Sheikh Rasel Ahmed(Nirob)
3 2022-06-07 11:11 PM      +880 1912-236428
4 2022-06-07 11:13 PM Sheikh Rasel Ahmed(Nirob)
5 2022-06-07 11:15 PM      +880 1912-236428
6 2022-06-07 11:17 PM Sheikh Rasel Ahmed(Nirob)
```

```
   Message      Positive      Negative      Neutral
2  Asscalamu Walaikum      0.0      0.0      1.0
3  Walicom as salam      0.0      0.0      1.0
4  ki obostha sobar kaj kmn kortecen      0.0      0.0      1.0
5  Now in class with Hafiz Bhai.      0.0      0.0      1.0
6  ki class      0.0      0.0      1.0
```

C:\Users\User\AppData\Local\Temp\ipykernel_5612\1075751284.py:7: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

C:\Users\User\AppData\Local\Temp\ipykernel_5612\1075751284.py:9: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

C:\Users\User\AppData\Local\Temp\ipykernel_5612\1075751284.py:11: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

```
data["Neutral"] = [sentiments.polarity_scores(
```

```
In [51]: data.head()
```

```
Out[51]:
```

	Date	Time	Author	Message	Positive	Negative	Neutral
2	2022-06-07	11:10 PM	Sheikh Rasel Ahmed(Nirob)	Asscalamu Walaikum	0.0	0.0	1.0
3	2022-06-07	11:11 PM	+880 1912-236428	Walicom as salam	0.0	0.0	1.0
4	2022-06-07	11:13 PM	Sheikh Rasel Ahmed(Nirob)	ki obostha sobar kaj kmn kortecen	0.0	0.0	1.0
5	2022-06-07	11:15 PM	+880 1912-236428	Now in class with Hafiz Bhai.	0.0	0.0	1.0
6	2022-06-07	11:17 PM	Sheikh Rasel Ahmed(Nirob)	ki class	0.0	0.0	1.0

In [52]:

```
data.info
```

Out[52]:

```
<bound method DataFrame.info of          Date      Time      Author \
2  2022-06-07  11:10 PM  Sheikh Rasel Ahmed(Nirob)
3  2022-06-07  11:11 PM                +880 1912-236428
4  2022-06-07  11:13 PM  Sheikh Rasel Ahmed(Nirob)
5  2022-06-07  11:15 PM                +880 1912-236428
6  2022-06-07  11:17 PM  Sheikh Rasel Ahmed(Nirob)
..      ...      ...      ...
314 2022-07-09   9:33 PM  Sheikh Rasel Ahmed(Nirob)
315 2022-07-09  11:11 PM                Roni-vai-hr Ve
316 2022-07-10   7:26 AM                +880 1912-188308
317 2022-07-12  10:29 AM  Sheikh Rasel Ahmed(Nirob)
318 2022-07-12  10:51 AM                +880 1796-336003

      Message      Positive      Negative \
2          Asscalamu Walaikum          0.0          0.0
3          Walicom as salam          0.0          0.0
4      ki obostha sobar kaj kmn kortecen          0.0          0.0
5      Now in class with Hafiz Bhai.          0.0          0.0
6          ki class          0.0          0.0
..      ...      ...      ...
314          EID MUBARAK          0.0          0.0
315      <Media omitted>          0.0          0.0
316      <Media omitted>          0.0          0.0
317 I got Project is there anyone available to do ...          0.0          0.0
318          yes          1.0          0.0

      Neutral
2          1.0
3          1.0
4          1.0
5          1.0
6          1.0
..      ...
314          1.0
315          1.0
316          1.0
317          1.0
318          0.0

[289 rows x 7 columns]>
```

In [53]:

```
data.describe()
```

Out[53]:

	Positive	Negative	Neutral
count	289.000000	289.000000	289.000000
mean	0.097554	0.022810	0.865789
std	0.229468	0.107607	0.267685
min	0.000000	0.000000	0.000000
25%	0.000000	0.000000	0.838000
50%	0.000000	0.000000	1.000000
75%	0.000000	0.000000	1.000000
max	1.000000	1.000000	1.000000

In [54]:

```
x = sum(data["Positive"])
y = sum(data["Negative"])
z = sum(data["Neutral"])
```

```
def sentiment_score(a, b, c):
    if (a > b) and (a > c):
        print("Positive 😊 ")
    elif (b > a) and (b > c):
        print("Negative 😞 ")
    else:
        print("Neutral 😐 ")
```

```
sentiment_score(x, y, z)
```

Neutral 😐

So, the data I used indicates that most of the messages between me and the other person are neutral. Which means it's neither positive nor negative.

Summary

So this is how we can perform the task of sentiment analysis of WhatsApp chat. WhatsApp is a great source of data for the task of sentiment analysis and every data science task based on natural language processing. I hope you liked this article on the task of WhatsApp chat sentiment analysis using Python. Feel free to ask your valuable questions in the comments section below.

Sheikh Rasel Ahmed

Data Science || Machine Learning || Deep Learning || Artificial Intelligence Enthusiast

In [55]:

```
# LinkedIn - https://www.linkedin.com/in/shekhnirab1
# GitHub - https://github.com/Rasel1435
# Behance - https://www.behance.net/Shekhrasel2513
```



© 2022 GitHub, Inc.

[Terms](#)

[Privacy](#)

[Security](#)

[Status](#)

[Docs](#)

[Contact GitHub](#)

[Pricing](#)

[API](#)

[Training](#)

[Blog](#)

[About](#)