# Twitter Sentiment Analysis

**Ritu Ramakrishnan**

02/27/2023

—

COMP261 Data Science

—

Pro. Jinzhu Gao

# Files

## Python Files

1. Scapper.py
2. tweet_sentiment.py
3. term_sentiment.py
4. frequency.py
5. happiest_state.py
6. top_ten.py

## Input Files:

1. tweets_data.json (generated tweets in json format)
2. data.json (file which was provided)
3. positive_tweets_data.json ( online data to test the term sentiment)
4. negative_tweets_data.json (online data to test the term sentiment)

## Output Files:

1. Output_tweets_sentiment.txt
2. Output_term_sentiment.txt
3. Output_frequency.txt
4. Output_Happiest_State.txt
5. Output_Top_Ten.txt

## Problem 1: Get Twitter Data

**Ans:** The **scraper.py** file generates the tweets in the json format.
These credentials from the twitter needs to be updated.

```
# credentials
consumer_key = "YOUR_CONSUMER_KEY"
consumer_secret = " YOUR_CONSUMER_SECRET"
access_token = "ACCESS_TOKEN"
access_token_secret = "ACESS_TOKEN_SECRET"
```

Figure: 1 Updating the twitter credentials

The tweets are collected and stored in the **tweets_data.json** file.
Attached the screenshot of code execution.

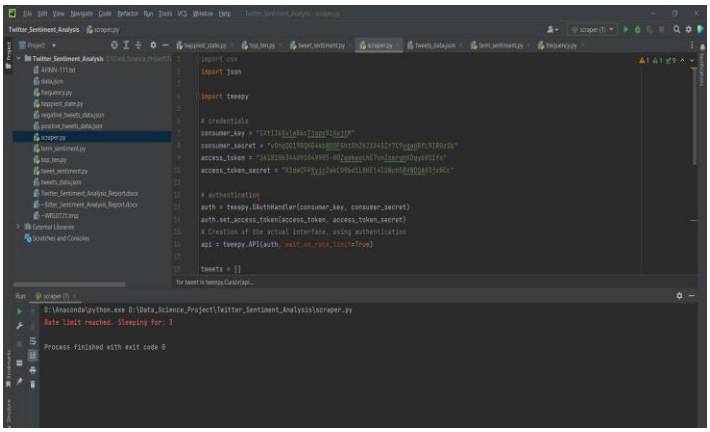

Figure: 2 Scraper.py execution

## Problem 2: Derive the sentiment of each tweet

**Ans:** To run this file run the below comment:

**$ python tweet_sentiment.py AFINN-111.txt tweets_data.json**

This **tweet_sentiment.py** file generates the sentiment for the words in the **AFINN-111.txt** file.
Attached the screenshot of the code execution:

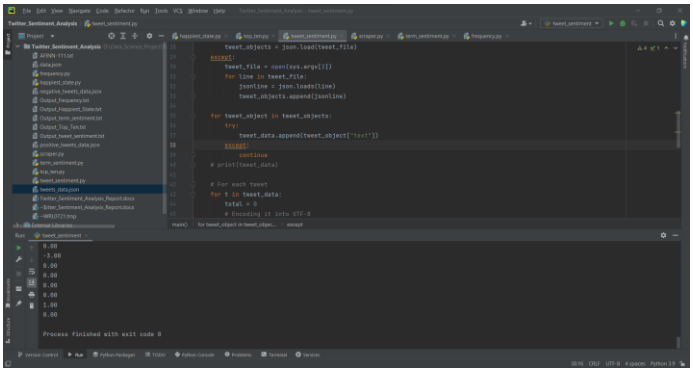

Figure: 3 tweet_sentiment.py execution

The output file for the execution is available in the **output_ tweet_sentiment.txt** file.

## Problem 3: Derive the sentiment of new terms

**Ans:** To run this file run the below comment:

**$ python term_sentiment.py AFINN-111.txt tweets_data.json**

This **term_sentiment.py** file generates the sentiments for the terms which are not in the **AFINN-111.txt** . I have eliminated the stop words and numerical values. For the testing purpose I have also used **positive_tweets_data.json** and **negative_tweets_data.json** for better understanding how the values are calculating.

Attached the screenshot of the code execution:

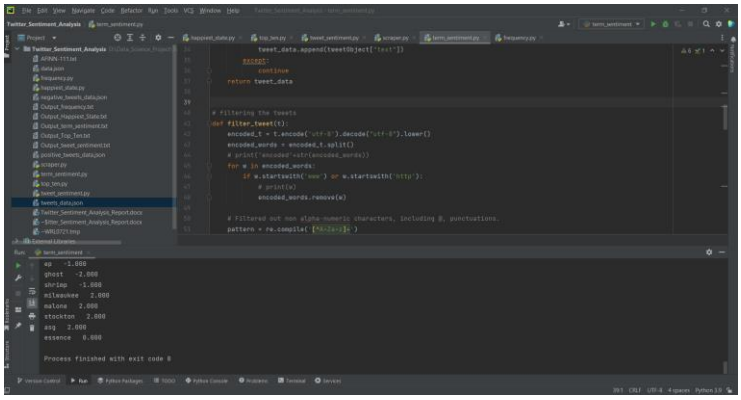

Figure: 4 term_sentiment.py execution

The output file for the execution is available in the **output_term_sentiment.txt** file.

## Problem 4: Compute Term Frequency

**Ans:** The **frequency.py** file is executed to compute the term frequency.

To run this file run the below comment:

**$ python term_sentiment.py tweets_data.json**

Attached the screenshot of the code execution:

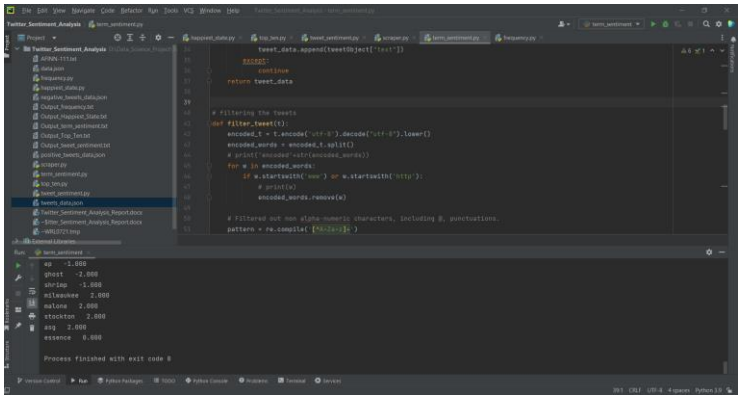

Figure: 5 frequency.py execution

The output file for the execution is available in the **output_frequency.txt** file.

## Problem 5: Which State is happiest?

**Ans:** The **happiest_state.py** file is executed to calculate the tweets which has the name of the happiest state as a string.
To run this file run the below comment:

**$ python happiest_state.py AFINN-111.txt tweets_data.json**

Attached the screenshot of the code execution:

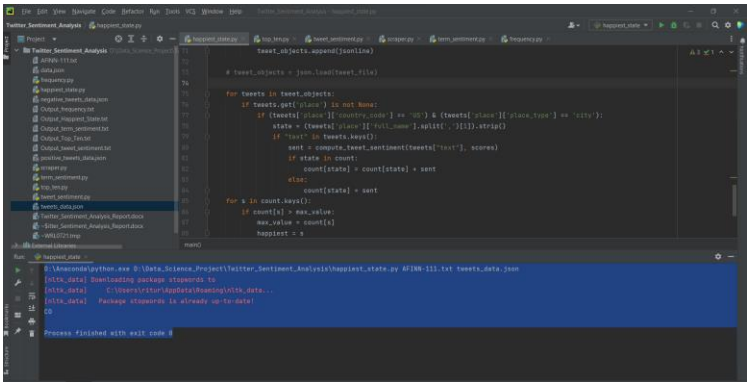

Figure: 6 happiest_state.py execution

The output file for the execution is available in the **output_Happiest_State.txt** file.

## Problem 6: Top ten hash tags

**Ans:** The **top_ten.py** that computes the ten most frequently occurring hashtags from the data.
To run this file run the below comment:

**$ python top_ten.py tweets_data.json**

Attached the screenshot of the code execution:

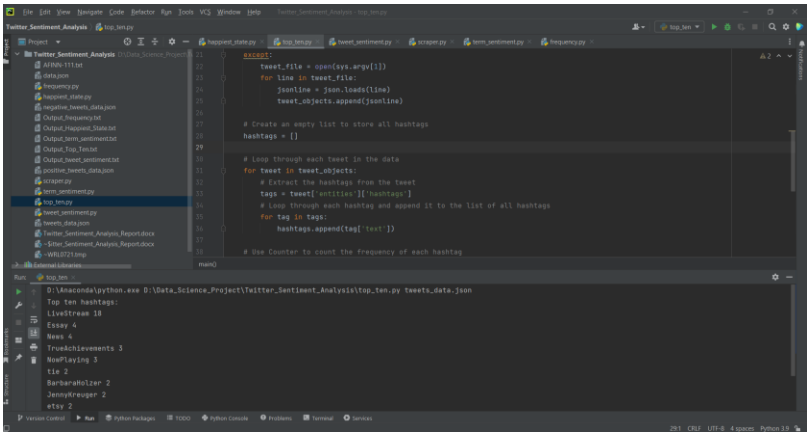

Figure: 7 top_ten.py execution

The output file for the execution is available in the **output_Top_Ten.txt** file.