Alexander Shaun Tan, Jenny Son

Python-Text Mining Project Write - Up

I. Project Overview

The data source we decided to use was Twitter. We wanted to run a sentiment analysis on a given input word, where we would use the module tweepy to obtain tweets and run natural language processing to obtain the general attitude Twitter has towards the term. This would allow us to determine the sentiments of the tweets related to the given input, specifically the average polarity and subjectivity of these words. We also wanted the option to compare one search term with another, so that users can see the difference in sentiments between the two.

II. Implementation

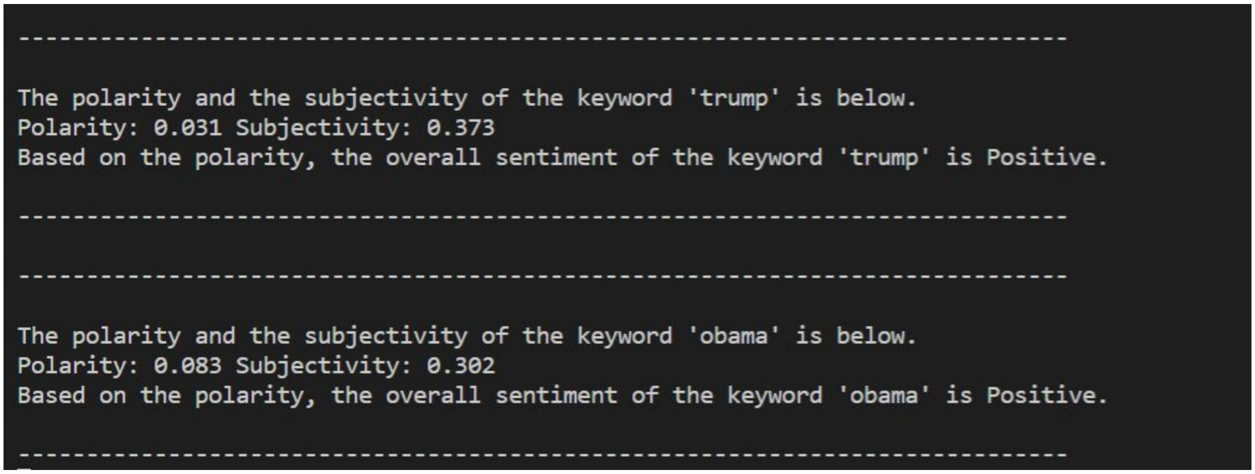
We first obtained the ability to dynamically get tweets from Twitter with the tweepy module. We limited the count to the latest 1000 tweets so that the sentiment is still relevant. Upon obtaining the capability to source current tweets, we were considering how to organize our data such that it could be easily run through sentiment analysis. We were deciding between creating a dictionary that allowed us to have the key as the user name, or a simple compilation of list that showed each individual tweet. We realized that if we wanted to call back to specific tweet in the future or locate the tweets of a specific user, it would be more effective to use a dictionary, especially since it would increase the readability of the code and make it easier to tie one tweet to one person. Had we continued with creating lists, we would be limited to just getting the overall sentiment of a total list of tweets. Hence, we created a dictionary that allowed us to have the key as the user name, while the value would be the specific tweet.

Upon placing the public tweets of a given search term into respective keys and values in the dictionary, we also wanted to clean the data within. We created a function that removed any unnecessary information such as the username of the twitter user, the attached URLs, as well as the Twitter-specific terms for retweets such as “RT”. By running a for loop on every word in the value of the dictionary, we were able to successfully create a clean dictionary ready for sentiment analysis.

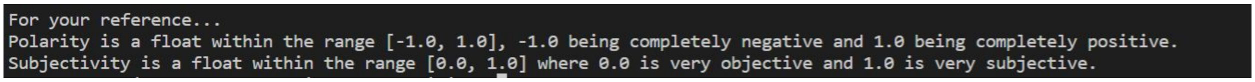
From this, we ran our dictionary of cleaned tweets under the sentiment function of the textblob module to obtain the analysis. We also wanted to give the option to compare a search term with another, so we allowed the search of up to two terms, with an accompanying visualization of the comparison with a bar chart with the module matplotlib. Our final output is a print of average polarity and subjectivity of up to two terms, which highlights the general attitude to the search terms from twitter, with a supplemented bar chart that shows a better visualization. To supplement this information, we also implemented a ‘Positive’ or ‘Negative’ threshold for polarity and a description of what subjectivity depicts. This is so that users of this code can understand what the numbers from the sentiment analysis means.

III. Results

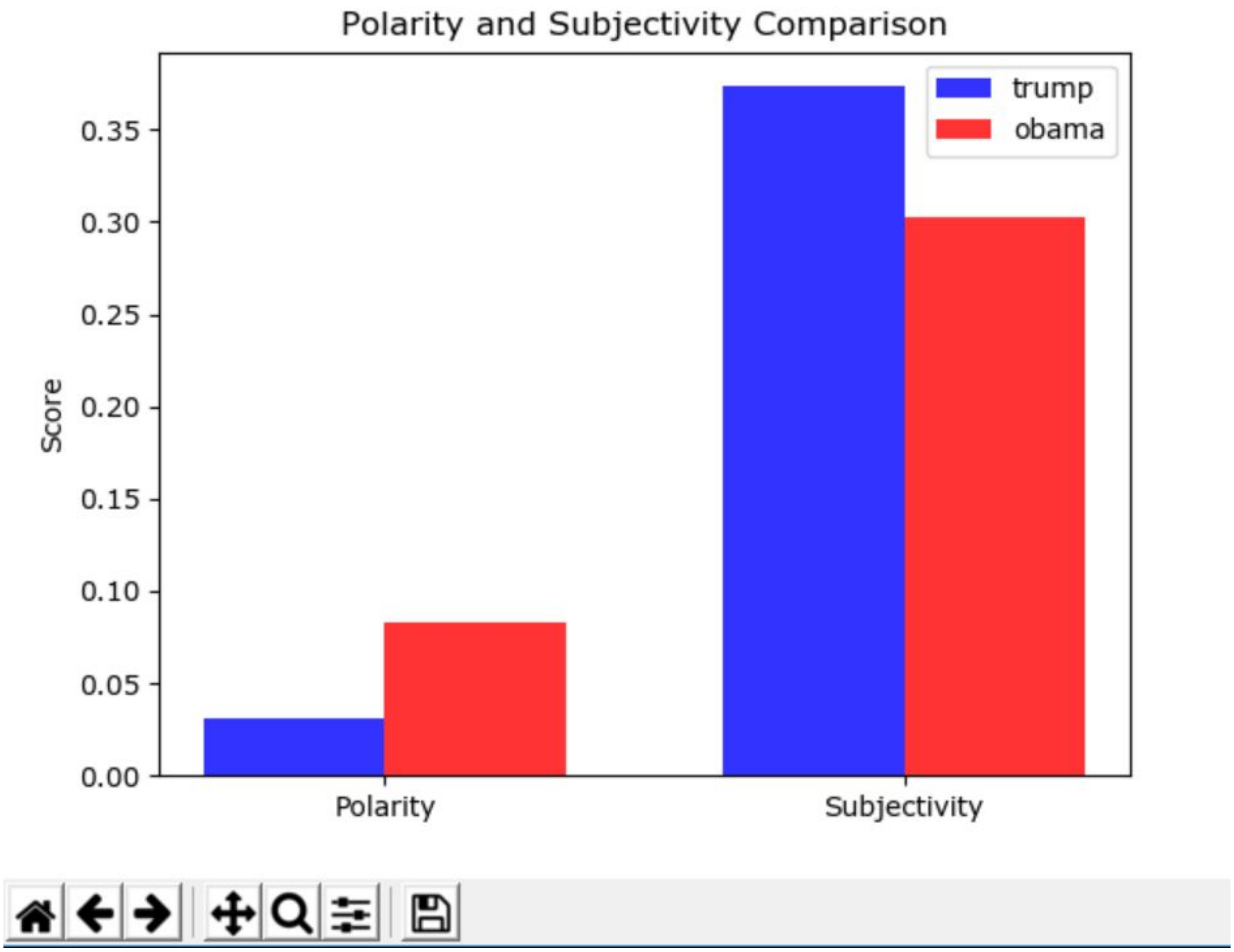
To test our code, we conducted a few searches to obtain the average sentiment towards certain terms. Firstly, we wanted to compare the search term of Trump and Obama to see the twitter attitude towards the two seemingly political rivals before. Upon searching and comparing the sentiments of the two terms from tweets, we interestingly got the results depicted in the figure below.



```
-----------------------------------------------------------------------------

The polarity and the subjectivity of the keyword 'trump' is below.
Polarity: 0.031 Subjectivity: 0.373
Based on the polarity, the overall sentiment of the keyword 'trump' is Positive.

-----------------------------------------------------------------------------

-----------------------------------------------------------------------------

The polarity and the subjectivity of the keyword 'obama' is below.
Polarity: 0.083 Subjectivity: 0.302
Based on the polarity, the overall sentiment of the keyword 'obama' is Positive.

-----------------------------------------------------------------------------
```

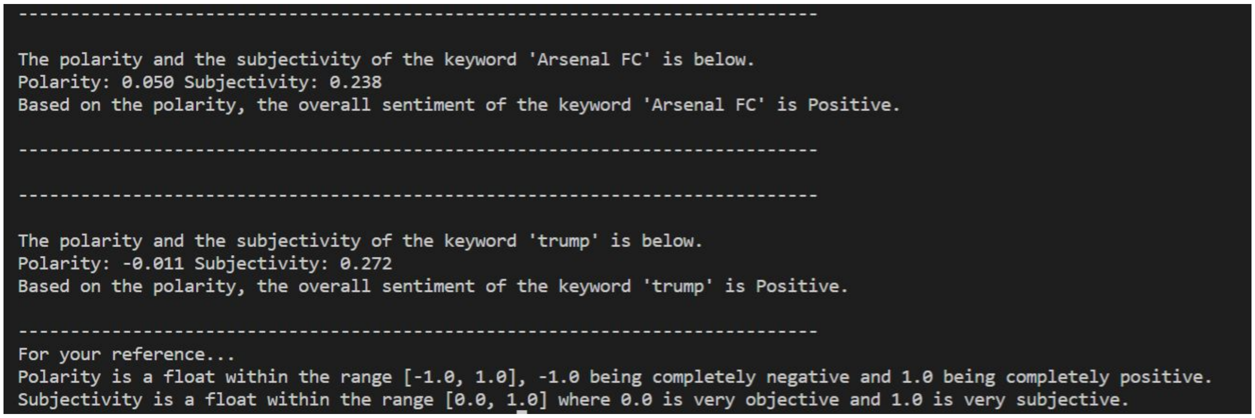


```
For your reference...
Polarity is a float within the range [-1.0, 1.0], -1.0 being completely negative and 1.0 being completely positive.
Subjectivity is a float within the range [0.0, 1.0] where 0.0 is very objective and 1.0 is very subjective.
```

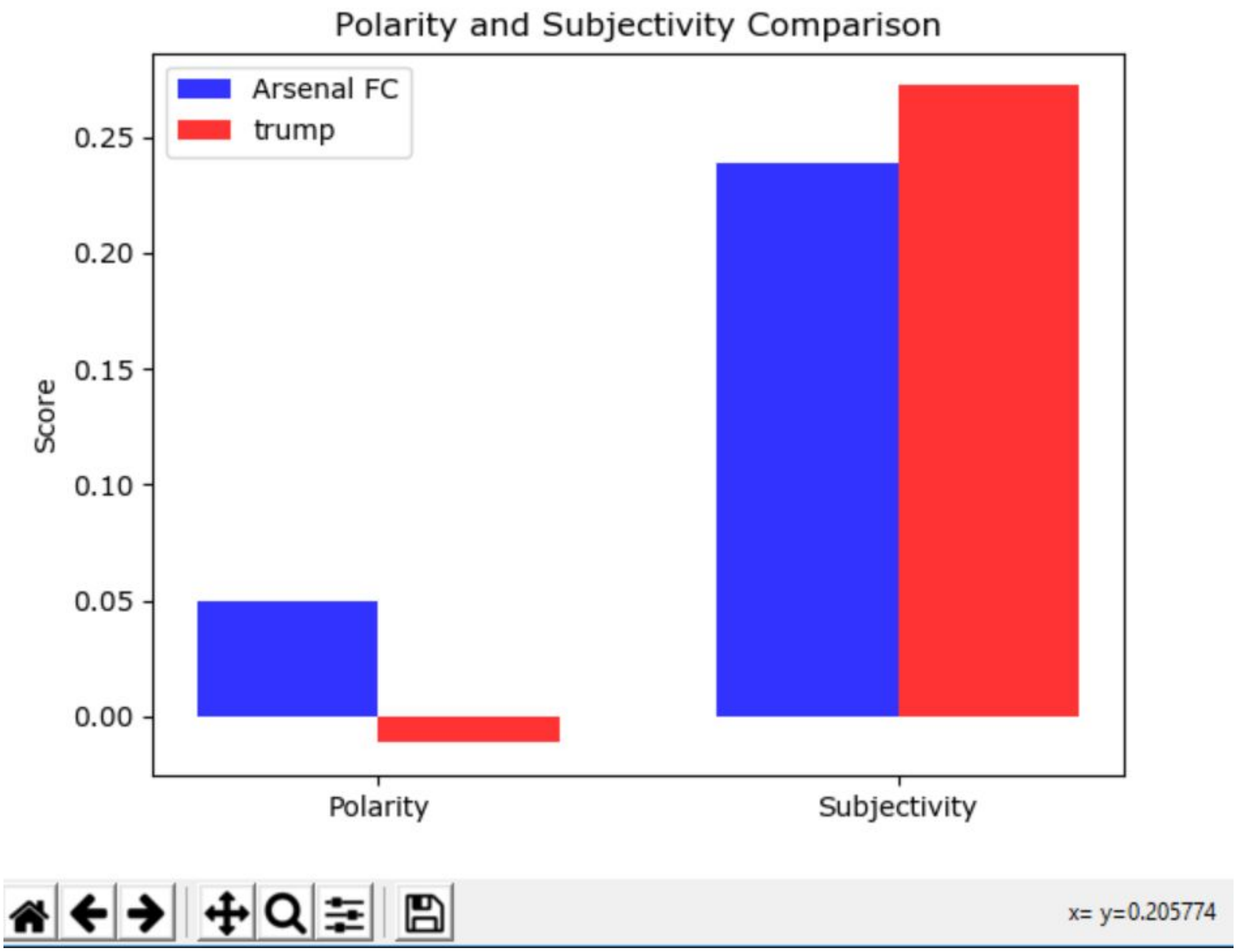


From here, it is evident that overall, both presidents have positive sentiments, but it is interesting to note that Obama has a higher positive polarity than trump. Furthermore, our analysis also highlights the importance of subjectivity by explaining whether the sentiments of a specific search term is objective, or subjective. In this example, it is clear that Trump is a generally more subjective term compared to Obama. This means that public opinion affects the sentiment of Trump more. We realized from this then that this code would be good when comparing the sentiments of two polarizing inputs.

Our code also verifies the overall sentiment towards a term that resonates with a lot of positive attitudes. For example, we searched the average sentiment for a football team for the term, ‘Arsenal FC’, where we hypothesized that there would be mostly positive sentiments dominated by tweets from the club’s fans. This is seen in the exhibit:

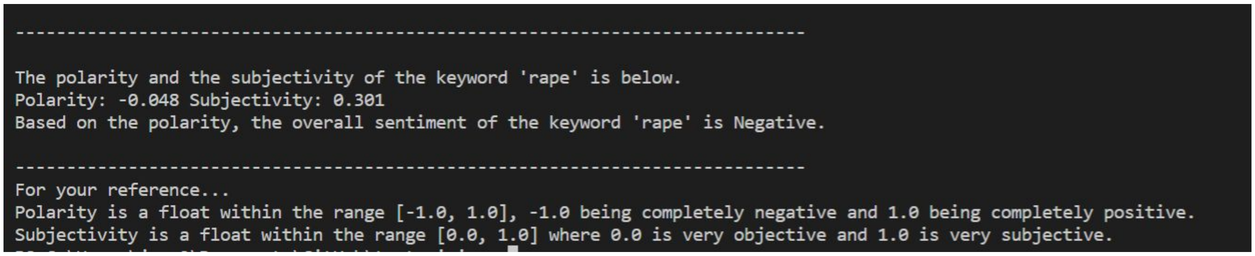


```
-----------------------------------------------------------------------------

The polarity and the subjectivity of the keyword 'Arsenal FC' is below.
Polarity: 0.050 Subjectivity: 0.238
Based on the polarity, the overall sentiment of the keyword 'Arsenal FC' is Positive.

-----------------------------------------------------------------------------

-----------------------------------------------------------------------------

The polarity and the subjectivity of the keyword 'trump' is below.
Polarity: -0.011 Subjectivity: 0.272
Based on the polarity, the overall sentiment of the keyword 'trump' is Positive.

-----------------------------------------------------------------------------
For your reference...
Polarity is a float within the range [-1.0, 1.0], -1.0 being completely negative and 1.0 being completely positive.
Subjectivity is a float within the range [0.0, 1.0] where 0.0 is very objective and 1.0 is very subjective.
```



In this output, it is clear that an input that generally relates to less political or debatable contexts has a higher polarity than a term that evokes more opinion or discussion, like that of trump.

Lastly, we searched the term ‘rape,’ which is a term that is clearly much more negative than the terms tested previously.



```
--------------------------------------------------------------------------

The polarity and the subjectivity of the keyword 'rape' is below.
Polarity: -0.048 Subjectivity: 0.301
Based on the polarity, the overall sentiment of the keyword 'rape' is Negative.

--------------------------------------------------------------------------
For your reference...
Polarity is a float within the range [-1.0, 1.0], -1.0 being completely negative and 1.0 being completely positive.
Subjectivity is a float within the range [0.0, 1.0] where 0.0 is very objective and 1.0 is very subjective.
```

From the exhibit, it is clear that our code highlights the negative connotations for a word like ‘rape,’ which shows that most tweets on Twitter do not resonate with the term and its meaning.

IV. Reflection

Overall, we believe we effectively came up with a code that provides some utility in obtaining and comparing sentiment over search terms. We were able to work collaboratively and share ideas such that our code could reach our end goal. We conducted this successfully by first, meeting in-person to discuss overall goals and steps to reach this goal. From there, we delegated the individual coding work, where Shaun performed the code on sentiment analysis, and Jenny worked on cleaning the data set and placing it in the appropriate data type. We, however, made sure to communicate our work on a daily basis to ensure that we were on the same page. One issue that arose when we were working separately was the update of github. We had several github issues where our new codes not being pushed appropriately. Due to the github issue, we tried our best to work together in the same space so that we would not have to deal with such a problem.

While working, we aimed to achieve a stretch goal of allowing the users to search for another keyword and providing a comparative bar chart that shows differences in polarity and subjectivity between two keywords. We wanted to create more visualizations for our project but we were not able to do so because of the tight time limit. In total, however, we strongly believe that this project is a success, where there are insights and some value given from the data analysis.