**Twitter Text Capture and Analysis**

**Objective:**

This activity deals with text analysis of data pulled from Twitter. Using this data, we answer a few important questions that helps us to get better insight into the collected data and derive meaning from it.

**Procedure:**

1. Download the twitter script from the following link & look at the script.

<https://drive.google.com/open?id=0B3n4oQctrRw6cFJIM3N4Q3ZMZFE>

* 1. How does the program work? It allows the user to search for a string between a chosen time period. The user also selects the number of tweets to retrieve for the string that is being searched for. Some information is displayed per tweet retrieved, such as: the user who posted it, when it was created, the text, the followers, and the retweet count.
  2. How do you think you can use this code? We could use this for data analytics and studies.
  3. Can you think of different scenarios where this code could be used for data collection? For example, we could pull tweets from a specific time period (such as a during a crisis) in order to find patterns in human behavior. It could also be used for businesses – perhaps a business wants to search for their name to see what kinds of thing customers (or other people) are saying about them/their products/services.

1. Add your twitter API key in the script. You can generate one using <https://apps.twitter.com/> .
2. Run twitter.py , input the required details to generate the output file
3. The output file contains user name, time, tweet, retweet\_count, and the followers for the user.

**Challenge:**

1. Now write a program to read the file generated. Your program should get the value of n (the number of records to be displayed) from the user, and generate the following output and write them into separate files.
   1. The top n users who have tweeted the most for the entire timeline.
   2. The top n users who have tweeted the most for every hour.
   3. The top n users who have the maximum followers.
   4. The top n tweets which have the maximum retweet count.

**Guidelines:**

1. Attach a zip file containing text document, source code as well as the output generated for the above scenarios for top 10 users and include how to run your program in the word document.
2. Check in your code in github and specify the github repository link in the word document. You can refer the steps below to create a repository in github.

Instructions for running the code:

* + - 1. Make sure you have run the *twitter.py* first so that you have a text file to start with.
      2. In the *twitter\_file\_reader.py*, the console prompts you to enter a text file name. You don’t need to add the entire location (as long as the text file is in the same folder as the *twitter\_file\_reader.py*), only the name of the file (also exclude the .txt extension). For example, if the file from step 1 was “cheesecake” and produced a “cheesecake.txt”, simply enter “cheesecake” when prompted.
      3. Next, the console prompts you to enter *N* – this is how many records will be found and displayed for each output (top n users who have tweeted the most, etc…)
      4. And that’s it! The output file will be saved with the same name plus “\_output” added to the end. For example, “cheesecake.txt” will have the output of “cheesecake\_output.txt” and will be in the same folder as the original.

**Github repository link:** <https://github.com/jelloh/twitter-analysis>

**Creating a repository in GitHub**

1. Go to<https://github.com/>
2. Select Start a project.
3. Specify a repository name in the text box. Click create repository.Use Git bash to move to the directory that contains the folder to be moved from local machine to Github remote repository.
   1. **To initialize git:** git init
   2. **To add the files to git:** git add README.md
   3. **Committing the files to repository:** git commit -m "first commit"
   4. **Add it to your github repository:** git remote add origin**“your github repository name”**
   5. **Push it to the repository:** git push -u origin master

**Future enhancements:**

1. Sentiment analysis based on some keywords that reflect the emotions related to the tweets.
2. Location based searching to identify local preferences and trends.