Bo Zhang
01063214

Task 1: Write a Python program that extracts 1000 unique links from Twitter. Also note that you need to verify that the final target URI is unique.

Algorithm:

1. Connecting to Twitter Streaming API and downloading data. Use the script from `http://adilmoujahid.com/posts/2014/07/twitter-analytics/`, but save the output to "output.txt" instead of printing on the screen.
2. Reading and parsing the data. Also use the script from `http://adilmoujahid.com/posts/2014/07/twitter-analytics/`, but only save the texts of tweets to the list instead of saving everything.
3. Extracting links from the list with tweets texts. Also use the script from `http://adilmoujahid.com/posts/2014/07/twitter-analytics/` to get the original links. Then open original links to get the final URIs from the response.
4. Removing the duplicated links.
5. Removing the "unreal" URIs and spam URIs. Delete URIs starting with `https://twitter.com/` to remove links back into twitter itself. And since short URIs tend to be spams, deleting URIs less than 50 bytes can remove spams.
6. Save the links to "links.txt".

Source code:

twitter_streaming.py

twitter_ExtractLinks.py

Results: links.txt

Task 2: Download the TimeMaps for each of the target URIs. Create a histogram of URIs vs. number of Mementos.

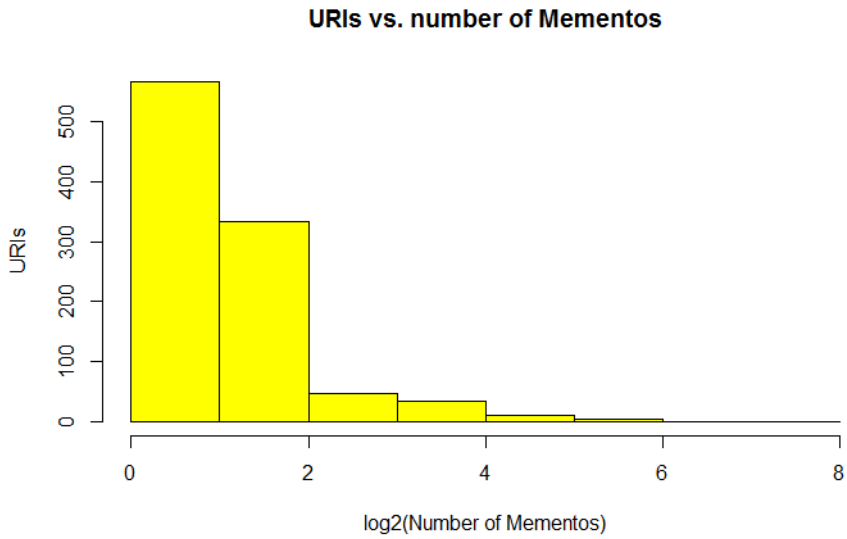
Algorithm:

1. Open "links.txt" and read links from the file 1 by 1.
2. Get response from the ODU Memento Aggregator of this link and save it into a BeautifulSoup object.
3. Traverse all descendants of the BeautifulSoup object.
4. For each descendant, search if there is any Memento in it.
5. Count how many times it was found.
6. Save the links and numbers of Mementos to "data_hist.csv".

Source code: twitter_ComputeMementos.py

R code: histogram.R

Results: data_hist.csv



Task 3: Estimate the age of each of the 1000 URIs using the "Carbon Date" tool. For URIs that have > 0 Mementos and an estimated creation date, create a graph with age (in days) on the x-axis and number of mementos on the y-axis.

Algorithm:

1. Open "links.txt" and read links from the file 1 by 1.
2. Get response from the "Carbon Date" tool of this link and save it into a BeautifulSoup object.
3. Extract the Estimated Creation Date from it.
4. Traverse all Estimated Creation Dates.
5. For each date, extract the date string and calculate the age (days between the date and now).
6. Save the links and ages to "ages.txt".
7. Merge the file and "data_hist.csv", remove URIs with no Mementos or date estimate and save it as "data_scatter.csv". (I used EXCEL in this step)

Source code: twitter_ComputeAges.py

R code: plot.R

Results:

ages.txt

data_scatter.csv

total URIs: 1000

no mementos: 567

no date estimate: 10