

Assignment 6

CS 432

MIRANDA SMITH

Question 1

Use D3 to visualize your Twitter followers. Use my twitter account ("@phonedude_mln") if you do not have ≥ 50 followers. For example, @hvdsonp follows me, as does @mart1nkle1n. They also follow each other, so they would both have links to me and links to each other.

I chose to use my friend's twitter account 'mango964' because he had fewer followers than 'phonedud_mln' with only 222 followers. I used the program 'getFollowers.py' that I created for a previous assignment to gather his followers.

```
import twitter

outfile = open("twitterFollowers.txt", 'w')
# Variables that contains the user credentials to access Twitter API
ACCESS_TOKEN = '824794440409706496-CGUoqhc2pFk4lOcb1NVG1OuubwaCq2p'
ACCESS_SECRET = 'Mgd7wctkwI3nNUL6RHD4Cqjg9HVngccRrynoWfEsTUz1J'
CONSUMER_KEY = '72haUMMgwOXNNGXXz4G0N7B9m'
CONSUMER_SECRET = 'zSx800UtH7OUTkVUMw2uORB39o1aP59uZgmXMW08AZpoFkNgJQ'

api = twitter.Api(consumer_key=CONSUMER_KEY,
                  consumer_secret=CONSUMER_SECRET,
                  access_token_key=ACCESS_TOKEN,
                  access_token_secret=ACCESS_SECRET,
                  sleep_on_rate_limit=True)

user = api.GetUser(screen_name="mango964")
userdict = user.AsDict()
userid = userdict["id"]

followerslist=[]
followersdata = api.GetFollowerIDsPaged(user_id=userid)
next_cursor = followersdata[0]
followerslist.extend(followersdata[2])

while next_cursor != 0 :
    followersdata = api.GetFollowerIDsPaged(user_id=userid, cursor=next_cursor)
    next_cursor = followersdata[0]
    followerslist.extend(followersdata[2])

outfile.write(str(userid)+ "\n")
for item in followerslist:
    outfile.write(str(item) + "\n")

outfile.close()
```

I created the program 'collectGraph.py' to collect the data between each pairing to see if one followed the other. I tried to implement it to collect the followers for each user and then compare to see which ones were in the graph but the time to run it would have been around the length of time needed to run this program which collects the relationship status between two users.

```

import twitter

outfile = open("links.txt", "w")

ACCESS_TOKEN = '824794440409706496-CGUoqhcZpfk41Ocb1NVG1OuubwaCqZp'
ACCESS_SECRET = 'Mgd7wctkwI3nNUL6RHD4Cqjg9HVngccRrynoWfEsTUz1J'
CONSUMER_KEY = '72haUMMgwOXNNGXXz4G0N7B9m'
CONSUMER_SECRET = 'zSx800UtH7OUTkVUMw2uORB39o1aP59uZgmXMW08AZpoFkNgJQ'

api = twitter.Api(consumer_key=CONSUMER_KEY,
                  consumer_secret=CONSUMER_SECRET,
                  access_token_key=ACCESS_TOKEN,
                  access_token_secret=ACCESS_SECRET,
                  sleep_on_rate_limit=True)

userList=[]
for line in open('twitterFollowers.txt', 'r'):
    line = line.strip()
    userList.append(line.strip())

#print(userList)

#see if two users have a relationship
#don't check the same people twice
for idx, source in enumerate(userList):
    if idx != (len(userList) - 1):
        for target in userList[idx + 1:]:
            try:
                result = api.ShowFriendship(source_user_id= source ,target_user_id = target)
                following = result["relationship"]["target"]["following"]
                print(following)
                follows = result["relationship"]["target"]["followed_by"]
                print(follows)
                source_name = result["relationship"]["source"]["screen_name"]
                target_name = result["relationship"]["target"]["screen_name"]

                if following:
                    print("TRUE")
                    outfile.write(str(source_name) + "\t" + str(target_name) + "\n")
                if follows:
                    print("TRUE")
                    outfile.write(str(target_name) + "\t" + str(source_name) + "\n")
            except Exception as err:
                print(err)
                print(source)

outfile.close()

```

I made sure to not collect data on the same relationship twice, to save time, since they give data on if the user is following or followed by the target user. I ran the program in its entirety on the original set of data and it took 3 days to run. I found out the data was bad due to a coding error on my part and had to run it again. I reduced the number of followers to 59, for a total of 60 nodes. I just chose the first 60 that were in the list and made sure to include the original user.

I made another short python program 'getFollowersName.py' to translate the user IDs that I had been working with into their screen name to be displayed in the graph. I forgot

to grab it when getting the connections and I was not going to run the program again to

```
#get user names from IDs
results = api.UsersLookup(user_id = userList)

for result in results:
    outfile.write(result.screen_name + "\n")

outfile.close()
```

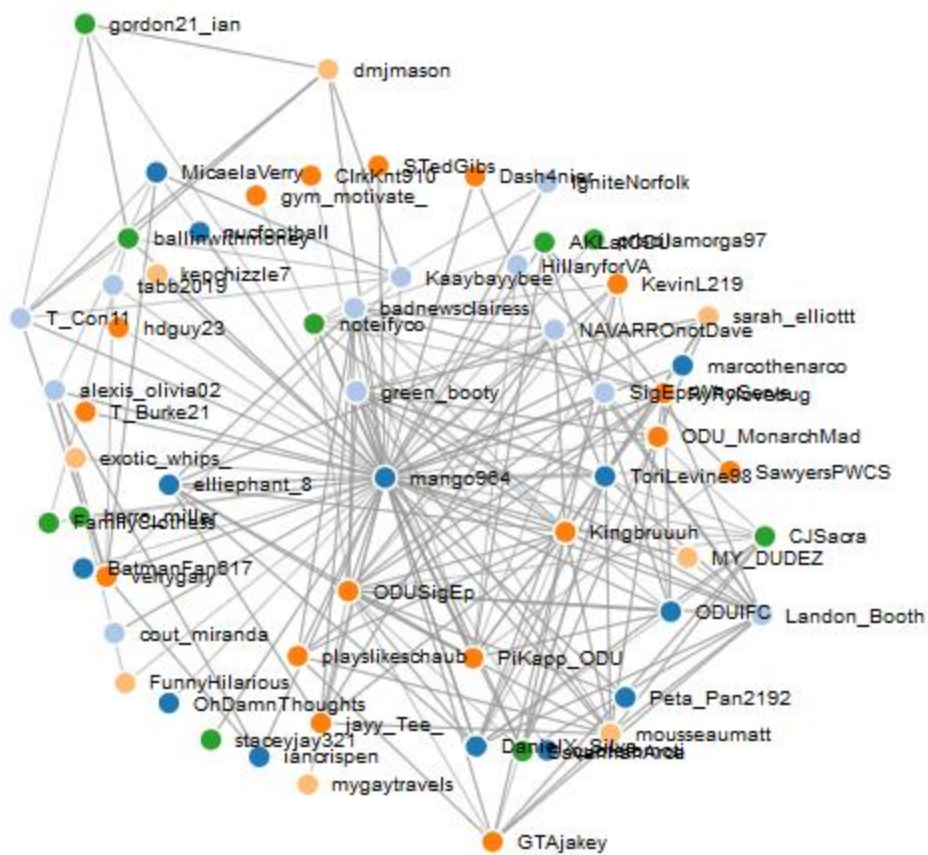
get them.

I ran the resultant link and user name files through 'toJson.py' to format it properly to be read by the d3 program. I assigned each node to a random group between 1 and 5 to be used for colors later.

```
#format the friendships and people from the twitter data into json
#so that the graphing program can read it
import json
from random import randint

data = []
node= {}
with open("userNames.txt", "r") as nodefile:
    for node in nodefile:
        node = node.strip()
        data.append({"id" : node, "group" : str(randint(1,5))})
data2= []
with open("links.txt", "r") as linkfile:
    for link in linkfile:
        link = link.strip()
        wordlist = link.split("\t")
        data2.append({"source" : wordlist[0], "target" : wordlist[1], "value" : "4"})
final = {"nodes": data, "links" : data2}
#create node dictionary
with open("data.json", "w") as fp:
    json.dump(final, fp, sort_keys = True, indent = 4)
```

I made a graph using d3 and svg to show an interactive directed graph to show who is following who in the small twitter sample. The program is too long to reasonably fit in this report but it is contained in 'index.html'. I got the data from the data.json file and linked each node by a name id instead of node number. I referenced the following graph to start off my graph and link them by names <https://bl.ocks.org/mbostock/4062045>. I used this to help add labels <http://stackoverflow.com/questions/13364566/labels-text-on-the-nodes-of-a-d3-force-directed-graph>. I also made small changes to space the nodes out further and add arrowheads to make it directed.



Here is a link to interact with the graph <https://rawgit.com/Miranda-C-Smith/cs532-s17/master/assignments/a6-solution/index.html> .