ASSIGNMENT 2

CS 432 Web Science

Mackenzie Kerchner

Mackenzie Kerchner

Contents

Mackenzie Kerchner

Problem 1

1. Write a Python program that extracts 1000 unique (collect more e.g., 1300 just in case) links from Twitter. Omit links from the Twitter domain (twitter.com).

Solution

```
class TwitterAuthenticator():

    def authenticate_twitter_app(self):
        auth = OAuthHandler(twitter_cred.CONSUMER_KEY, twitter_cred.CONSUMER_SECRET)
        auth.set_access_token(twitter_cred.ACCESS_TOKEN, twitter_cred.ACCESS_TOKEN_SECRET)
        return auth
```

I used [1] this link as the basis for twitter authentication and moved the actual keys to separate file to keep them hidden.

```
from tweepy import OAuthHandler
from tweepy import API
from tweepy import Cursor
import re
import requests
import twitter_cred

# import sys                    #
# reload(sys)                   #
# sys.setdefaultencoding('utf8') # some weird error with old formatting, fixed until i break it again

keepcount = int(1)  # how many tweets have been scrubbed through
```

These are all the modules used in the current version, the sys encoding is for an error with string handling that's been written out. I'm leaving it in as I'm willing to bet the error will reappear if I rewrite this code in the future.

```
# loads the url file into a list
initialUrlList = []  # this will hold all the links already in the list of links
infile = open("tweets.json", "r")
for line in infile:
    initialUrlList.append(line.strip())  # strip prevents extra unicode stuff, ex: /n for newline on each line
```

This section loads in any previously grabbed tweet links so newly grabbed links can be checked for duplicates.

```
if __name__ == '__main__':
    twitter_client = TweetGrabber()
    api = twitter_client.get_twitter_api()

twitter_client = TweetGrabber('MSNBC')  # this grabs from the 'tweets & replies' timeline
#twitter_client = TweetGrabber('NFL')  # this grabs from the 'tweets & replies' timeline
#twitter_client = TweetGrabber('FoxNews')  # this grabs from the 'tweets & replies' timeline

all_the_links = twitter_client.get_user_timeline_tweets(1000)
```

The number of tweets and user to scrub through is hard coded, it's easier to test quickly this way.

```
def get_user_timeline_tweets(self, num_tweets):
    tweets = []
    for tweet in Cursor(self.twitter_client.user_timeline, id=self.twitter_user).items(num_tweets):
        tweet_text = tweet.text
        link = re.findall("(?P<url>https?://[^\s'\"]+)", tweet_text)  # grabs links from tweet
        link = ','.join(link)  # i don't know how this was my best solution but here we are. turns single element list into a string

        global keepcount
        print keepcount
        keepcount += 1

        if link != [] and link != "" and link != '' and '\\u' not in link:  # disallows blanks added to link list
            try:
                tempint = 0
                fulllink = requests.get(link)  # note: this breaks when the url is already unshortened
                while "t.co/" in fulllink.url and tempint < 15:
                    fulllink = requests.get(fulllink.url)
                    tempint += 1
                print ("unshortened link: ", fulllink.url)

                if "t.co" not in fulllink.url \
                        and "twitter.com/" not in fulllink.url \
                        and fulllink.url not in tweets \
                        and fulllink.url not in initialUrlList:
                    tweets.append(fulllink.url)
                else:
                    print "bad link"
            except:
                print "error in link tweet parsing----------------------------------"  # hyper-advanced error detection

    print "------------------------------------------------------------------------------------------"
    print tweets
    return tweets
```

This is the main body of twitgrab.py. The findall is from [2] this link, formatting a findall for generic html links was huge pain until I found this one that had everything already covered.
I parsed a stream rather than using the search feature provided by the Tweepy api, and it scrubs for tweets from the specified user rather than by most recent. It resulted in some interesting (and annoying) results. The percentage of good links pulled from tweets varies significantly between users, MSNBC returns about 1 good link out of 10 tweets scrubbed, FoxNews returns nearly 4 out of 10. The error detection is rudimentary, but an enormous amount of links grabbed point to twitter.com, and this cuts them out rather quickly. The except only occurs in 2-3 tweets out of 1000, I haven't figured out what's causing this bigger error yet but since it's exceedingly rare I didn't think it necessary for now.

```
946
('unshortened link: ', u'https://twitter.com/i/web/status/1098538328952131584')
bad link
947
('unshortened link: ', u'https://twitter.com/i/web/status/1098529784274042881')
bad link
948
('unshortened link: ', u'https://twitter.com/i/web/status/1098526017361186816')
bad link
949
('unshortened link: ', u'https://twitter.com/i/web/status/1098523481782796288')
bad link
950
('unshortened link: ', u'https://www.msnbc.com/the-beat-with-ari/watch/why-mueller-could-ask-ag-barr-to-indict-trump-in-office-1445794883949?cid=sm_npd_ms_tw_ma')
bad link
951
('unshortened link: ', u'https://www.msnbc.com/morning-joe/watch/sen-durbin-does-sen-graham-really-want-to-investigate-doj-fbi-1445376579648?cid=sm_npd_ms_tw_ma')
952
('unshortened link: ', u'https://twitter.com/i/web/status/1098499579534815232')
bad link
953
('unshortened link: ', u'https://www.msnbc.com/morning-joe/watch/sen-durbin-does-sen-graham-really-want-to-investigate-doj-fbi-1445376579648?cid=sm_npd_ms_tw_ma')
bad link
954
('unshortened link: ', u'https://twitter.com/i/web/status/1098486247360790529')
bad link
955
('unshortened link: ', u'https://www.msnbc.com/rachel-maddow/watch/stone-faces-consequences-of-attacking-judge-at-thursday-hearing-1445892163531?cid=sm_npd_ms_tw_ma')
bad link
```

Above is the console output, note that the good links are being reported as bad because this is not the first time this has been run and these are duplicates.

```
1100  http://www.msnbc.com/rachel-maddow-show/trump-seems-unaware-his-plan-end-criminalization-homosexuality?cid=sm_npd_ms_tw_ma
1101  https://www.nbcnews.com/video/chaos-in-an-ohio-courtroom-as-man-attacks-his-attorney-1446002755750?cid=sm_npd_ms_tw_ma
1102  http://www.msnbc.com/rachel-maddow-show/trump-unreliable-narrator-about-his-own-presidency?cid=sm_npd_ms_tw_ma
1103  https://www.msnbc.com/rachel-maddow/watch/stone-faces-consequences-of-attacking-judge-at-thursday-hearing-1445892163531?cid=sm_npd_ms_tw_ma
1104  https://www.msnbc.com/morning-joe/watch/what-s-next-in-the-mueller-probe-1446044739569?cid=sm_npd_ms_tw_ma
1105  https://www.nbcnews.com/politics/meet-the-press/biden-s-final-hesitation-about-2020-what-it-could-mean-n973921?cid=sm_npd_ms_tw_ma
1106  https://www.nbcnews.com/politics/donald-trump/mike-pompeo-says-he-has-ruled-out-running-senate-seat-n973906
1107  https://www.nbcnews.com/politics/donald-trump/mike-pompeo-says-he-has-ruled-out-running-senate-seat-n973906?cid=sm_npd_ms_tw_ma
1108  https://www.msnbc.com/morning-joe/watch/sen-durbin-does-sen-graham-really-want-to-investigate-doj-fbi-1445376579648?cid=sm_npd_ms_tw_ma
1109  https://www.msnbc.com/all-in/watch/andrew-mccabe-tells-chris-hayes-the-president-may-be-compromised-1445824579513?cid=sm_npd_ms_tw_ma
1110  https://www.nbcnews.com/news/latino/human-rights-groups-say-deaths-venezuelan-protesters-appear-be-targeted-n973651?cid=sm_npd_ms_tw_ma
```

Above is the tweets.json file viewed in notepad++. Note in the full file, the first thousand results are from Fox News, as they gave the single largest group of good links.

References

[1] https://stackoverflow.com/questions/6399978/getting-started-with-twitter-oauth2-python

[2] https://stackoverflow.com/questions/839994/extracting-a-url-in-python

[] reference