CS 2310 Project Milestone 2

Yingze Wang

yiw32@pitt.edu

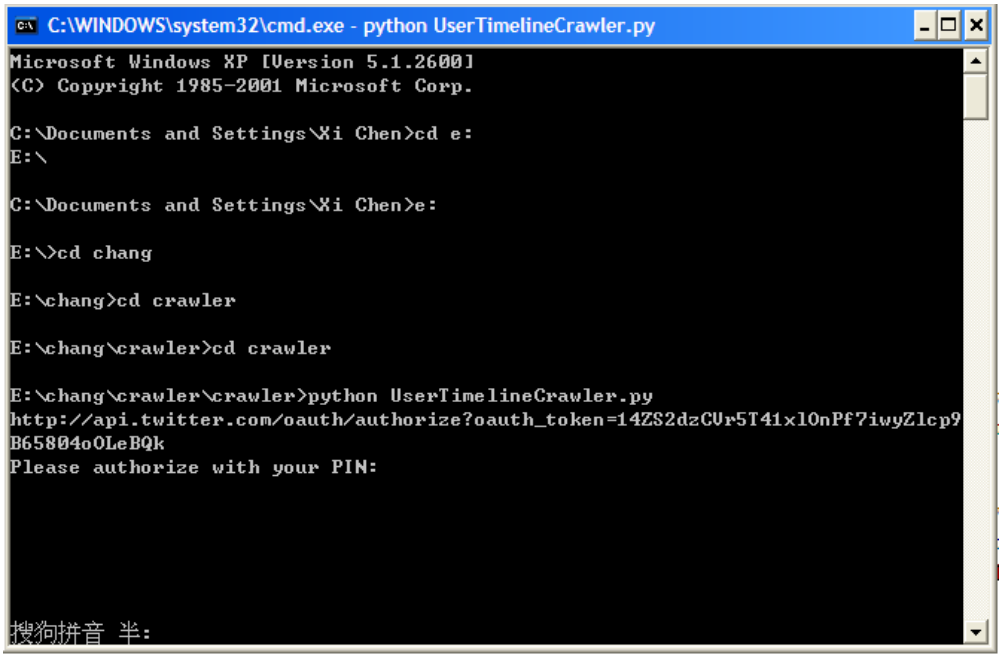
Twitter Web-Data Crawler

In this project, I have already developed a python crawler which can efficiently crawl the tweets data in a real-time and preprocess the raw data to extract the useful information I need including tweets text, user and time.

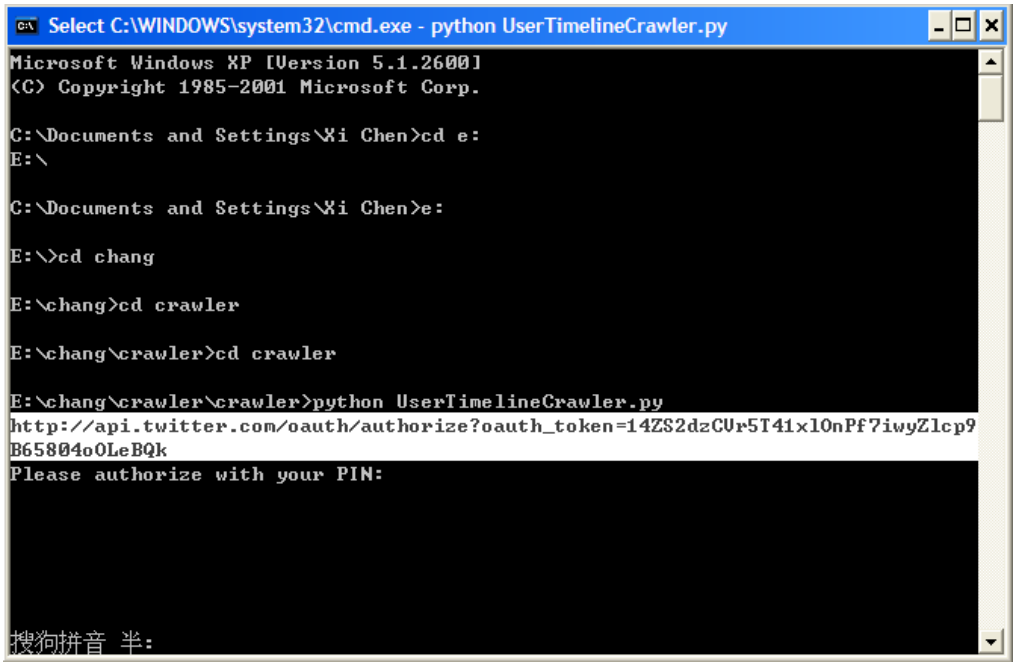
I. Crawling raw data from twitter

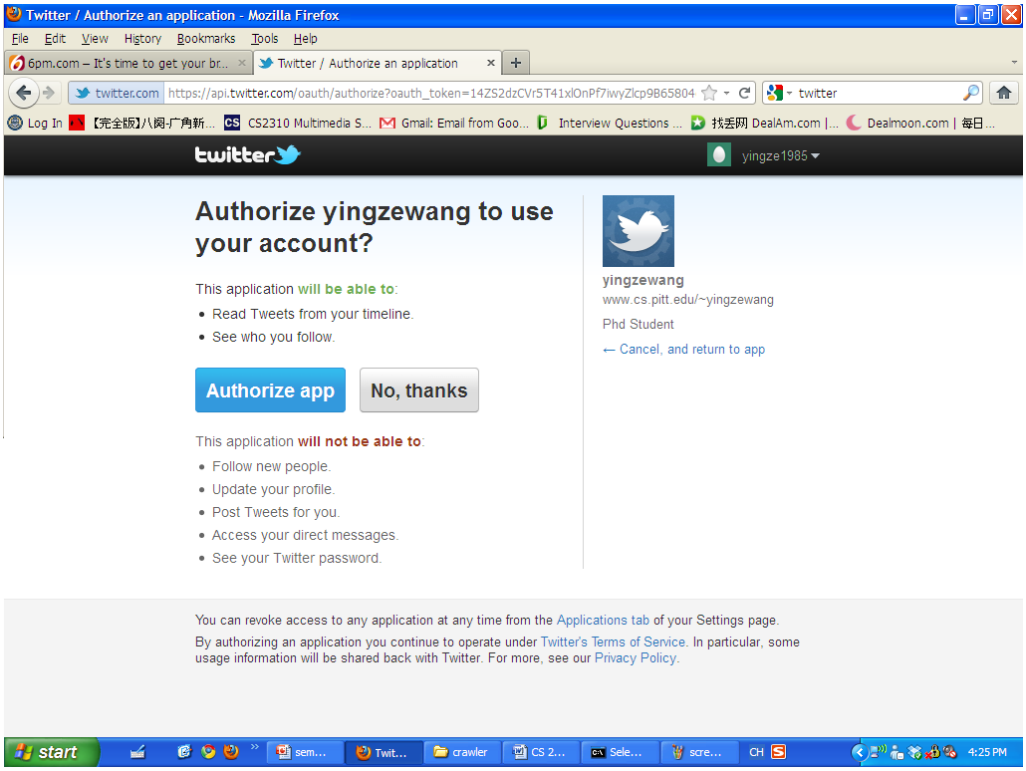
Firstly, I create the seed of nearly 1000 users who are the famous people or posting many tweets. Then for each user, the python program can crawl the tweets from this user file and save it as a txt. Each line is a tweets record, with the format (key: value). The screen shot of crawler is simple:

1. Open command window, start the program “UserTimelineCrawler.py”

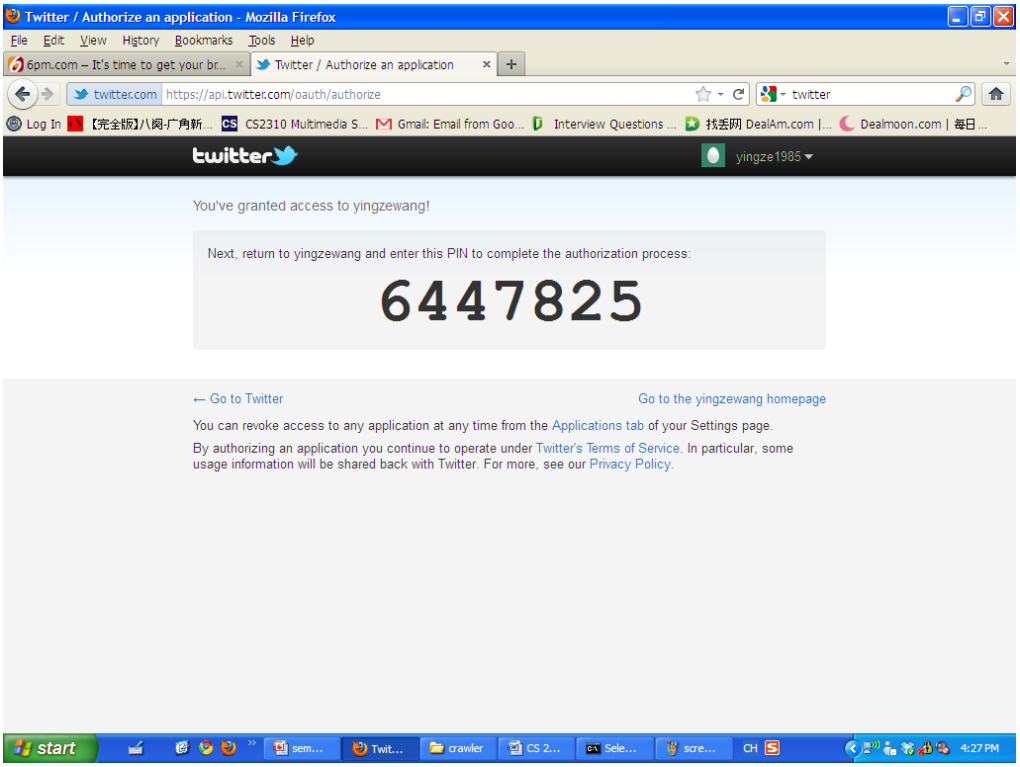


2. Due to the authorization process, you need to login the twitter. And Copy this unique url to the browser. Every time running, it will create a new unique url address.

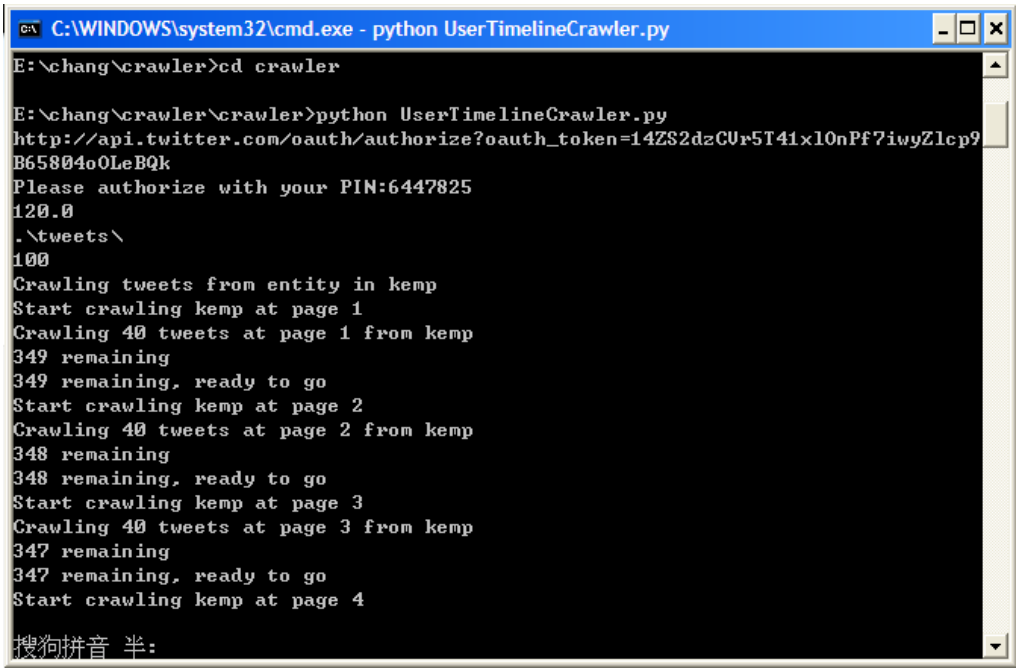




3. Click on “Authorize app” tab, twitter will offer an unique code to user and authorize the user crawl the data.



4. Then input this PIN number on the command window, then the program will begin to crawl the tweets. In this screen shot, we can see that the crawler is crawling the tweets from user named “kemp”.



Due to the privacy issue, twitter only allow the user to crawl about 100 data each time, then we need to wait about 1 hour to re-crawl again. The crawled data format is shown like:

{"favorited": false, "in_reply_to_user_id": 14779052, "contributors": null, "truncated": false, "text": "@brandonchicago Come join the fun. Plenty of room on the bandwagon!", "created_at": "Mon Oct 17 15:07:20 +0000 2011", ……}

For each user, the crawler crawl the whole tweets and create one txt file. Each line is one tweet record shown above.

II. Preprocess the raw data and select hot topics

There are many attributes in each tweet record, due to our need for data format, we are only interested in the “text”, “created_at”, username attributes. Thus, I use a python code to preprocess the data and extract these informations from raw data.

Then I use word count for each English word and filter out some words like “good” “Ok” which cannot be the topics. Finally we can select the interesting topic by this ranking. The ranking is shown like:

```
com 47159
google 35773
day 33834
twitter 31782
video 29641
media 29034
facebook 27962
top 24015
iphone 23532
awesome 21798
blog 20413
web 19678
app 17937
mobile 17415
apple 16615
youtube 14803
marketing 13829
que 13021
html 11637
seo 11434
fun 11417
try 11339
phone 10466
ipad 10150
www 10116
job 9936
bad 9836
san 9761
gutschein 9666
android 8881
```

……