

```

1
2 # coding: utf-8
3
4 # ## Non-subtweets Downloader Jupyter Notebook-in-Progress
5
6 # In[ ]:
7
8
9 import tweepy
10 import json
11
12
13 # ##### Set up access to the API
14
15 # In[ ]:
16
17
18 consumer_key, consumer_secret, access_token,
19 access_token_secret = open("../credentials.txt").read().
20 split("\n")
21
22 # In[ ]:
23
24
25 auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
26 auth.set_access_token(access_token, access_token_secret)
27
28 # ##### Specifically take advantage of built-in methods to
29 handle Twitter API rate limits
30
31 # In[ ]:
32
33
34 api = tweepy.API(auth, wait_on_rate_limit=True,
35 wait_on_rate_limit_notify=True, compression=True)
36
37 # ##### Find tweets with replies that do not claim it is a
38 subtweet
39
40 # In[ ]:
41
42
43 def first_tweet(tweet_status_object):
44     try:
45         return first_tweet(api.get_status(tweet_status_object.
46 in_reply_to_status_id_str, tweet_mode="extended"))
47     except tweepy.TweepError:
48         return tweet_status_object
49
50
51 def get_non_subtweets(max_tweets=1000000, query=(-subtweet
52 AND @ since:2018-03-01 exclude:retweets filter:replies")):
53     non_subtweets_ids_list = []
54     non_subtweets_list = []
55     i = 0
56     for potential_non_subtweet_reply in tweepy.Cursor(api.
57 search, lang="en", tweet_mode="extended", q=query).items(
58 max_tweets):
59         i += 1
60         potential_non_subtweet_original = first_tweet(
61 potential_non_subtweet_reply)

```

```

53         if not potential_non_subtweet_original.
in_reply_to_status_id_str and potential_non_subtweet_original
.user.lang == "en":
54         if potential_non_subtweet_original.id_str in
non_subtweets_ids_list or "subtweet" in
potential_non_subtweet_original.full_text or "Subtweet" in
potential_non_subtweet_original.full_text or "SUBTWEET" in
potential_non_subtweet_original.full_text:
55             continue
56         else:
57             non_subtweets_ids_list.append(
potential_non_subtweet_original.id_str)
58             non_subtweets_list.append({"tweet_data":
potential_non_subtweet_original._json, "reply":
potential_non_subtweet_reply._json})
59             with open("../data/other_data/non_subtweets.
json", "w") as outfile:
60                 json.dump(non_subtweets_list, outfile,
indent=4)
61                 print("Tweet #{0} was a reply to a non-
subtweet: {1}\n".format(i, potential_non_subtweet_original.
full_text.replace("\n", " ")))
62             return non_subtweets_list
63
64 # In[ ]:
65
66
67 non_subtweets_list = get_non_subtweets()
68 print(len(non_subtweets_list))
69

```