INSA – Project X

Twitter

Scrapping data from twitter (or currently X)

Abenezer Tamirat
10/4/2024

Table of Contents

1. Methods to scrap data from twitter

There are two ways to scrap data from twitter:

- **Using the standard API**:

Twitter provides API for developers to interact with and use twitter data. We can create developer profile on twitter developers portal and get access tokens to request data on behalf of ourselves and other users who authenticate us

- **Using third party libraries**:

Because of limitation and complexity of using the standard API there exist many libraries which extract data through other means. There are different ways this libraries work.

2. Using the Standard API

A. Authentication

- Before a developer makes an API call to X-API they need to create an app on the developer portal in order to get proper authentication tokens.
- An app is just a method of identifying an API call to X's API
- X API allows a developer to make API calls on behalf of a user or on behalf of an app itself.
- It has five types of tokens for authentication
    - API key and API secrete
        - Are a combination of two keys which are used to make a request on behalf of an app
    - Bearer token
        - Instead of using API key and API secrete each time we make API call to the X's API we can generate an app Bearer token using them once and use it instead of them
    - Access token and Access secrete
        - When a user allows our app to make API request on behalf of it they will give us access token and secrete

        - We use this keys to make API calls on behalf of any user
        - To get this we can use a login form generated using our app API and Secrete key or other methods
- In general we can request data from X's API as some X user or as an APP itself
- Authentication of X API is complex
- Types of authentication supported
    - OAuth 1.0
    - OAuth 2.0
        - App only
        - User
    - Basic authentication with email and password

B. Rate limits

- Rate limits for X API calls depend on subscription.
- Subscription starts from free, basic, pro account up to enterprise
- For free plan (user access token) we have 50 requests per day to GET, POST, and DELETE separately
- There is a separate rate limit when information is requested on behalf on an app
- Error message when limit reached
    - `{ "errors": [ { "code": 88, "message": "Rate limit exceeded" } ] }`
- Headers used to get rate limit info
    - `x-rate-limit-limit`: the rate limit ceiling for that given endpoint
    - `x-rate-limit-remaining`: the number of requests left for the 15-minute window
    - `x-rate-limit-reset`: the remaining window before the rate limit resets, in UTC epoch seconds

C. X API objects

- X api have endpoints for the following objects
    - POSTS

- USERS
- LISTS
- SPACES
- MEDIA
- PLACES
- POLLS

D. Fields

- Each object has default fields and we can request additional fields
- There are added new fields for every object like metrics, conversation id

E. Request endpoint

- For the version 2 of the X-api we use https://api.x.com/2/ end point
- For example:
  - https://api.x.com/2/users/by/username/abenezer_tamirat

3. Using third party libraries

Third party libraries scrap data bypassing the standard API through different methods:

- It could be through requesting for pages on behalf of a user and parsing the HTML
- Or they could mimic a web browser and intercept responses used to hydrate the HTML that a website loads

There are some drawbacks of using these libraries:

- Account could be blocked when the activity is suspicious
  - Therefore we have to mimic a human user by using random delays in our search

A. Libraries to scrap data

There are a lot of python libraries that are used to scrap data from twitter. There are officially supported libraries which interact with the twitter API and there are libraries that scrap data from twitter using unconventional methods.

- Officially Twitter Supported libraries:
    - **tweepy**
    - **twarc**
    - **python-twitter**
    - **TwitterAPI**
    - **twitterati**
    - **twitter-stream.py**
    - **twitivity**
    - **PyTweet**
    - **tweetkit**

- Not officially Supported libraries:
    - **Scrapy**
    - **Beautiful Soup**
    - **Selenium**
    - **Twint**
    - **Twikit**

4. Libraries I used

- To scrap data from twitter I used a python library called **twikit**, which uses scrapping (parsing HTML) pages.
- Steps involved
    - Authenticated a client with credentials
    - Make a search request with specific query
- Currently I only scrap tweets and users. Below are the attributes for users and tweets.

- The format to store the fetched data is **.csv** format for the time being. In the future integration with SQLite will be done until we found central database hosting from Ethiotelecom or any other hosting
- Storage of these data is crucial not to lose the data fetched for future use and training an artificial intelligence

5. Attributes of the data scrapped

I. Tweets	
community_note	Community note attached with z tweet
created_at	Date when it was tweeted
edits_remaining	How many edits remain for the tweet
favorite_count	The count of users who favorites it
full_text	The full text of the tweet
Hashtags	Hashtags separated with comma
has_card	Whether the post has a card for a hyperlink
has_community_notes	Bool for community note
Id	The id of the tweet
in_reply_to	The tweet id in replay to
is_quote_status	Whetere the tweet has quoted another tweet
is_translatable	Whether the tweet is translatable
Lang	language
Media	Links of Media it contain separated by comma
Place	The place attached with the tweet
Poll	Wheter the tweet contains a pool
possibly_sensitive	Whether the tweet is sensitive
Quote	The tweet id it quotes
quote_count	How many quotes it contains
reply_count	The number of comments or replies
retweet_count	The number of retweets
retweeted_tweet	If the tweet is retweeted it contains the

	original tweet id
thumbnail_title	The thumbnail title
thumbnail_url	The thumbnail url
Urls	Urls in the post separated by comma
user	The id of the author of the tweet (can be used to get user info from users table)
view_count	How many views it has
view_count_state	Is it possible to view the view count?

II. Users

can_dm	Whether the user can be messaged
can_media_tag	Whether the user can be tagged
created_at	When the user joined twitter
default_profile	Whether the user edited their profile
default_profile_image	Whether the user changed their profile picture from default
description	The description of the user
description_urls	The urls in the description
fast_followers_count	
favourites_count	
followers_count	
following_count	
has_custom_timelines	
id	
is_blue_verified	
is_translator	
listed_count	How many lists the user is in
location	The location of the user
media_count	Total media count of the user
name	The name of the user
normal_followers_count	Followers who are not premium users
pinned_tweet_ids	The tweet id if the user pinned a tweet
possibly_sensitive	Some accounts post sensitive content
profile_banner_url	

protected	Whether the account is private
statuses_count	How many posts the user made
translator_type	Whether the user is a translator (new twitter feature that uses human to translate messages)
url	The url of the user
urls	Other urls of the user separated by comma
verified	Whether the user is verified
want_retweets	Whether the user wants retweets

5. References

Twikit package — twikit documentation. (2024). Readthedocs.io. https://twikit.readthedocs.io/en/latest/twikit.html

Twitter API v2 tools & libraries. (2024). X.com. https://developer.x.com/en/docs/x-api/tools-and-libraries/v2

Twitter API Documentation. (2024). X.com. https://developer.x.com/en/docs/x-api