TWITTER API

All the large, well-known social networks and hundreds of international companies generate business thanks to the REST system (Representational State System), and Twitter, Instagram, YouTube and Facebook, were not going to be left behind.

When we speak of a REST system, we are referring to any interface between systems that use HTTP to obtain data and generate operations on those data in all possible formats, such as XML and JSON.

Without REST APIs, horizontal growth would be much more complicated. As for Twitter, its REST API allows you to read and write Twitter data; in other words, it can be used to create new tweets, read user profiles and the data of followers (among other data from each profile), since it identifies the various Twitter applications and the users who register using OAuth authentication and authorization. The replies from the Twitter REST API are in JSON format.
In addition to the REST API, the public Twitter API has an API for *streaming* that provides access to a high volume of tweets with low latency. Most developers mix and combine both APIs to generate their own application.
Basic features of the Twitter API

- **The Twitter API has four main "objects": Tweets, Users, Entities and Places.**
- **It has daily restrictions for calls and changes in the API to protect Twitter from abuses. Specifically, the restriction is set up by the user, or better said, by a user access token. The frequency restrictions are divided into 15-minute intervals and all the evaluation criteria require authentication so unauthenticated calls cannot be made to the API.**
- **The API is based on HTTP (over SSL), so the processes that require a specific HTTP method will return an error if the request is not correct.**
- **There are specific parameters for requests to the API, generated paging and library restrictions to adapt API operation to this social network.**

HOW TO ACCESS TWITTER'S API USING TWEEPY

Tweet datasets are an extremely desirable corpus for aspiring data scientists to analyze and perform models on. Tweets are by nature short-form and contain diverse and relevant topics, making it an excellent dataset for sentiment analysis. It is also a great dataset to analyze user engagement since tweet timestamps are available.

While there are many existing Twitter datasets out there, they will be predefined for you. I am a believer of first defining the problem you're interested in, and then figuring out a way to get that data. This is where it is beneficial to access Twitter API you get the type, volume and 'newness' only an API can provide.

WHAT IS TWEEPY?

An easy to use Python library for accessing the Twitter API.

Straight from the Tweepy website: https://www.tweepy.org/

The Twitter API exposes dozens of HTTP endpoints that can be used to retrieve, create and delete tweets, retweets and likes. It provides direct access to rich and real-time tweet data, but requires having to deal with a lot of low-level details.

Tweepy is an open source package that allows you to bypass a lot of those low-level details. Twitter's developer website has great documentation I recommend exploring to get example responses and see the type of data you are able to

access. Tweepy's documentation will furthermore provide code snippets and some basic documentation for the Tweepy module.

GETTING TWITTER API AUTHENTICATION

Twitter API uses OAuth, which is an open authorization protocol to authenticate requests. You will need to create and configure your authentication credentials to access Twitter API. As promised, this is a step-by-step guide so follow along!

Step 0: Open a Twitter account. If you already have a Twitter account, skip this step

Step 1: Apply for a developer account

Step 2: Create an Application

Step 3: Get your authentication details