

TWITTER ANALYSIS

Find potential influencer

GOAL: Find potential influencers on twitter that allow you to make your account / company known and grow.

INITIAL IDEA: create a web app in which you enter your account name and recreate your network on twitter by analyzing your followers and for each of them the corresponding followers. Through centrality measures, find the network hubs and analyze them.

REAL PROJECT: since building the complete network of followers requires a high computational cost, the project was divided into two parts:

1. iterative web app where you can analyse the current influencers of any twitter account
2. prediction of possible Influencers for a specific account (Masi Wines)

Twitter Analyzer

GOAL: web app that analyse Twitter accounts performance and find influencers, so accounts that allow to be more visible.

STEPS:

1. Data gathering
2. Data Analysis
3. Web App creations

1. DATA GATHERING

Requirements: Sign up for a twitter developer account <https://developer.twitter.com/en/apply-for-access>.

To scrape Twitter Data I've used **Tweepy API**, an open-source python package that provides a way for developers to communicate with the Twitter API. Twitter levies a rate limit on the number of requests made to the Twitter API.

- 1.1. Extrapolate tweets that mentioned the insert Account
 - Setting the authentication credentials
 - Instantiating the API
 - Creating API object to search last user-mentions tweets
 - Save all tweets

2. DATA ANALYSIS

2.1 FIND INFLUENCERS

■ Create directed weighted graph $G(V,E)$:

- V is the set of users,
- E set of links: there is an edge between the user and the mentioned user and also I add edges between mentioned users in the same tweets.
- Edge weight: to discern highly active non-influential users from the influencers I add a weight to edges as a function of their retweet account (x). Not to permit this effect to be overpowering I use the logarithmic function. $W = \log(x + 1)$

To build the graph I've used Networkx library (<https://networkx.org>).

■ Find influencer throw Eigenvector centrality: computes the centrality for a node based on the centrality of its neighbors.

- This measure allow to discern the low influence tweeters from the high, so bot account will have low centrality score

2.2 RETRIEVE SOME ACCOUNT'S DETAILS

■ Creating API User Object to search account details like number of followers, number of friends, the date on which it signed up

■ Calculate the average Tweets per day rate of that account using the Twitter User Object value `created_at` that holds the creation date of the account and `statuses_count` know how many Tweets that account has published.

■ Find the most used hashtag

2.3 WORDCLOUD

A word cloud is a collection, or cluster, of words depicted in different sizes. The bigger and bolder the word appears, the more often it's mentioned within a given text and the more important it is. With this technique is it possible to find target arguments of tweets.

2.4 ACCOUNT TIME SERIES SENTIMENT

Apply sentiment analysis to user-mentions tweets understanding how the account is perceived by others during the time.

■ Calculate sentiment to each tweet

■ Plot the average sentiment by year

3. WEB APP

I've use the open-source app framework Streamlit to create an interactive web app. You can only click the link below and insert any Twitter account name!

STRUCTURE OF CODE:

■ `network_influencer.py` : class to scrape data, create the graph and analyse it

■ `app_twitter_analyzer.py` : Streamlit code of web app

TRY IT: streamline run `app_twitter_analyzer.py`

Masi Wines potential Influencer

GOAL: predict potential influencers for Masi Wines account, that could be important for marketing campaign and to the expansion of its network.

STEPS:

1. Take the list of Masi followers (max 5000)
2. For each follower scrape its followers
3. Build the network of followers
4. Discover TOP 10 potential influencer using centrality measures, to be more precisely each measure has it's definition of "importance"
 1. In degree centrality: find influencer who has the higher number of followers
 2. Closeness: the more central profile are, the more quickly they can reach other accounts.
 3. Eigenvector: extension of degree centrality that keep track of neighbors importance. A high eigenvector score means that a node is connected to many nodes who themselves have high scores.
 4. PageRank: it's an improvement of degree-centrality. It takes into account the fact that not everyone known by a well known person is well known. To mitigate this problem, one can divide the value of passed centrality by the number of outgoing links (out- degree) from that node.
5. Understand potential influencer using Twitter Analyser

ADVERTING ! Tweepy as a rate limit, to be precisely only 900 requests/15 minutes. So this gathering it's computational expensive and I decide to stop at 5000 followers.

The directed MasiWines graph $G(V,E)$ has:

■ $|V| = 200868$

■ $|E| = 243247$

Applying the different centrality measures we can see that the influencer find are all different by which ones MasiWines has now (shows with Twitter Analyser).

At this point is necessary to understand if the potential nodes showed could be interesting for the account insert. Is it possible to analyse it with web app **Twitter Analyser**.

NOT CODED:

1. I would like to visualise the graph with Gephi but the network is very large and every time I try to process it the software crashes
2. I've tried to used Monkeylearn to understand the potential influencer twitter users (inspired by this code https://blog.twitter.com/developer/en_us/a/2015/guest-post-understanding-users-through-twitter-data-and-machine-learning.html) but due to request month limit I was unable to complete the code

VIEW THE RESULTS: streamline run app_full_analysis.py