Social Media Analytics - Project Report

Portaluppi Alessandro mat. 816090
Testa Luca mat. 816000

***Abstract*—The aim of this project is to analyse twitter stream data regarding to presidential American elections in 2020. Two candidates are Donald Trump and Joe Biden.**
In particular, this project provide a proposal for data retrieval using twitter API, and a R library for tweets stream, an application of sentiment analysis and social network analysis. Here are the results.

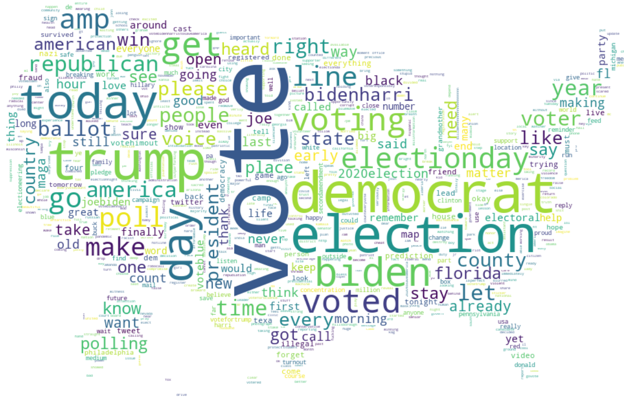

Figure 1. Most used words

I. Introduction

On 4th November 2020 the American elections were held, and we think this was a great occasion to analyse twitter stream data and comparing the sentiment to the real results of these elections.
First of all, is necessary to obtain Twitter API keys, from the official Twitter developers website: the free account is very restricted, using a single API key a developer can make a small number of queries per day. On the other hand, using a simple stream (query-filtered), we managed to get an huge number of tweets, so we think to develop an R script and start it for some hours during the election day.

II. Stream data from Twitter

As already mentioned, we start with implementing an R script for twitter streaming. The library used is *twitterR*.
Before running this script, we made an 'hashtag-based selection' with the aim of filter only tweets about election day: *stream_tweets* function allows to select only tweets that contains a particular expression or expressions.
After an accurate analysis, we chose the following hashtags: *#election, #vote, #politics, #trump, #usa, #elections, #democrat, #biden, #donaldtrump, #democrats, #america, #joebiden, #republican, #voteblue, #conservative, #voting, #democracy*.
The used function, has the particularity of getting JSON data, contains a large number of information about a tweet, much more than those that were really useful for our purpose. Moreover, JSON isn't a good language for data analysis in pandas, and the obtained JSON files had a very large dimension (up to 10gb), so for this reasons, we decide to convert JSON files into CSV files, keeping only 7 columns: datetime, tweet text, author username, author userid, location, followers and verified.
After this, our dataset was suitable for subsequent analysis.

III. Data cleaning and preparation

A. Text Cleaning

We start with a dataset containing about 500'000 tweets, posted in a period of about 5 hours. The first thing we need to do is manage relevant hashtags and tags. In particular we transform candidates' hashtags into simple text e.g. *#biden* -> *biden*, *#donaldtrump* -> *trump*. This step will be necessary when we estimate subject, to whom it refers, a tweet.
After this, we create two columns in a dataset containing respectively all hashtags and all mentions (tags) that the tweet contains, and so we remove all urls, useless for our analysis. The resulting dataframe has the following shape:

Table I

datetime	text	location	userid	screen_name	followers	verified	hashtags	mentions
.....	.....	.....	.....	.....	.....	....	[.....]	[.....]

After these preliminary steps, we focused on text cleaning and normalisation: first of all we use the function *tokenize* from the class *TweetTokenizer* (package nltk). This particular tokenizer is suitable for this type of text, handling also tags and hashtags.
The second steps was stop-words and punctuation removing, lower-casing and lemming, necessary for a better sentiment analysis. All of these features are available in package 'nltk'.

B. Location handling

For our purposes, keeping the larger number of record with a coherent value of location was of primary importance, in particular referred to the country state in the Unites States.
For this reason, we focused more on location normalisation, following this procedure:

- Keeping only tweets that has location field
- Obtain a dataset (uscities) containing the major cities in the US, and its state

- Search in 'uscities' a city placed in field location, and replace it with the corresponding state
- Using *OpenRefine* tool to try normalize remaining states
- Using Jaro–Winkler similarity for evaluate states written in different ways, and replace them with the normalized form: this is a simple algorithm to calculate similarity between two strings, taking into account matching characters, number of permutation and length of both strings. In our case, we consider only similarity values greater then 0.85.
- Remove all tweets with no-american location, and denormalized.

Doing this, all tweets in dataset matched with a location corrisponding to an US state.

IV. Sentiment Analysis

Starting with cleaned dataset, we can apply some techniques regarding sentiment analysis.

A. *Tweets' subject estimate*

The first step we need to implement is the estimate to whom the text refers (Donald Trump or Joe Biden). Doing this with a great precision is a very difficult task, so we decide to adopt a simpler way in order to impute text's subjet.

In particular, the strategy was counting how many times was present in text words referred to Biden's topic (biden, joe, democrats, democrat, bidenharris) and Trump's topic (trump, donald, republicans, republican, maga).

The greatest number of times that these words were written, established the subject of the tweet. If this number was the same, subject was imputed as 'undefined'. Here the results obtained using this method.

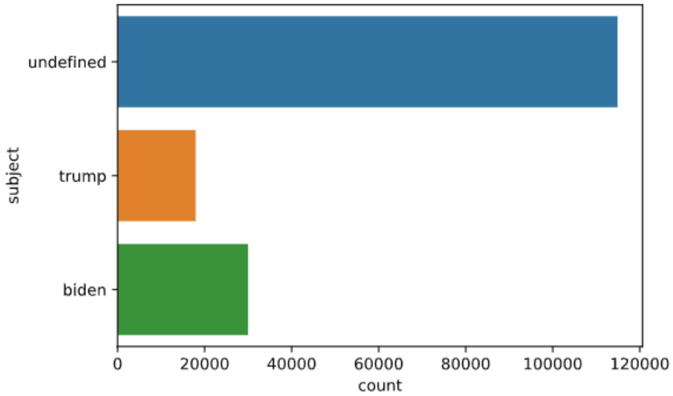

Figure 2. Subject count

As we can see, we have lots of documents that we aren't able to estimate subject, but in any case we proceed without these, reducing a lots the dataset's size.

B. *Sentiment analysis and emotions score*

In order to perform sentiment anaysis, we use Afinn score to assign a polartity score (positive or negative) for each tweet, relating to a candidate.

The AFINN lexicon is a lexicon-based approach that assigns words with a score that runs between -5 and 5, with negative scores indicating negative sentiment and positive scores indicating positive sentiment. [1]

The whole text is evaluated word by word, and the overall score of a sentence is calculated taking into account all word's score. After a normalisation step, we obtain, for each tweet a polarity value that runs between -1 (completely negative) and 1 (completely positive).

With the aim of obtaining a more complete sentiment analysis value, we decide to compute also a different metric present in literature, that is TextBlob polarity score.

TextBlob is an NLP tool for python language, that include also a sentiment lexicon to give polarity (between -1 and 1) and subjectivity (between 0 and 1) of a sentence.

The second part was regarding emotional affect measure for each tweet: to doing this, we used *NRCLex* library.

NRCLex(or NRCLexicon) is an MIT-approved PyPI project by Mark M. Bailey which predicts the sentiments and emotion of a given text. The package contains approximately 27,000 words and is based on the National Research Council Canada (NRC) affect lexicon and the NLTK library's WordNet synonym sets[2].

For our purpose, to each tweet was assigned a score regarding these emotions: trust, fear, anticipation and anger.

V. Social Network Analysis

The first thing was done was to extract from the dataset the date, the country, the user id, the clean text, the hashtags and the mentions made in the tweet.

Before moving on to true community detection, we also decided to delete multi-edges, loops and only keep nodes with a degree greater than three to avoid parsing groups of people that cannot be defined as communities because of too small degree.

Because the high number of nodes (90 thousand) it was decided to do a social network analysis only for the two states with opposite ideas: Texas and California. Thus, from the American dataset, only the rows belonging to the two states mentioned above have been selected.

Once the selection was made, the two graphs were initialized as: *screen_name* belonging to the nodes and *mentions* belonging to the connecting node.

Betweenness and degree correlation were calculated for both states. We have choice to use betweenness because in this type of analysis it seemed important to underline how a node must be defined central because it influences the flow around a system.

We use degree correlation to show if, in the presence of many famous people cited, an assortative, disassortative or random network was created.

[1] https://www.tidytextmining.com/sentiment.html

[2] https://www.kaggle.com/getting-started/196520

Thanks to the value of the betweenness, the top 10 nodes were chosen to represent the hubs of our graph.

Finally we partitioned the communities thanks to the use of modularity. Networks with high modularity have dense connections between the nodes within modules but sparse connections between nodes in different modules. The difference between the actual number of edges between node v and w and the expected number of edges between them is: $A_{v,w} - \frac{d_v d_w}{2E}$.

We also tried to use a "node centric" method based on the search of cliques. Obviously it was not possible to use having about 2 thousand nodes. The computational level would have been $2^{2000} - 1$.

TEXAS ANALYSIS

After calculating the modularity: *0.629*, we calculated the betweenness for the whole network. Obtained these values, for each single node, first ten were saved in order to identify those who were the "hubs" of this network:

Screen Name	Value
IvankaTrump	0.245
MaryLTrump	0.217
hypnoticOMG	0.206
RealJamesWoods	0.180
JoeBiden	0.102
LoriSums	0.096
Motivatedtweet	0.082
BetoORourke	0.079
DonaldJTrumpJr	0.076
no_silenced	0.051

Modularity has allowed us to divide the nodes into communities, to color the nodes belonging to the same community and to make them more or less large based on the value of the betweenness.

To plot the graph we used the ForceAtlas2 algorithm, used for graphing purposes from the well-known Gephi software, it is considered one of the best for its scalability and strength performance with networks up to 10,000 nodes.

ForceAtlas2 is a force directed layout: it simulates a physical system in order to spatialize a network. Nodes repulse each other like charged particles, while edges attract their nodes, like springs. These forces create a movement that converges to a balanced state. [..] Newman proposes an unbiased measure of this type of collective proximity, called "modularity". Noack has shown that force-directed layouts optimize this measure: communities appear as groups of nodes. [3] This network seems to be random with a degree correlation coefficient of *-0.069*.

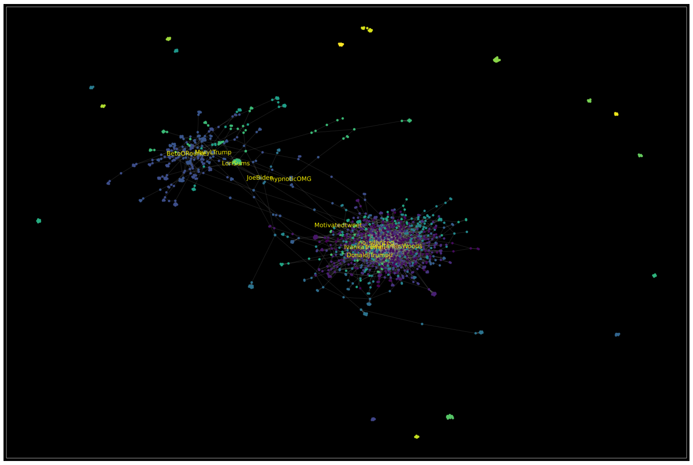

Figure 3. ForceAtlas2 with Texas

CALIFORNIA ANALYSIS

Also this network seems to be disassortative with a degree correlation coefficient of *-0.14*.

We calculated betweenness for this graph as well, keeping the top 10 characters as hubs:

Screen Name	Value
JoeBiden	0.204
RealJamesWoods	0.175
DebraMessing	0.143
ChristineS_1970	0.124
DonaldJTrumpJr	0.098
kymburleigh	0.094
SYMMETRY_11	0.090
MikeOkuda	0.085
realDonaldTrump	0.081
HuffPost	0.080

Also in this case it was decided to use ForceAtlas2 combined with modularity (communities color) and betweenness (nodes size) to plot the graph.

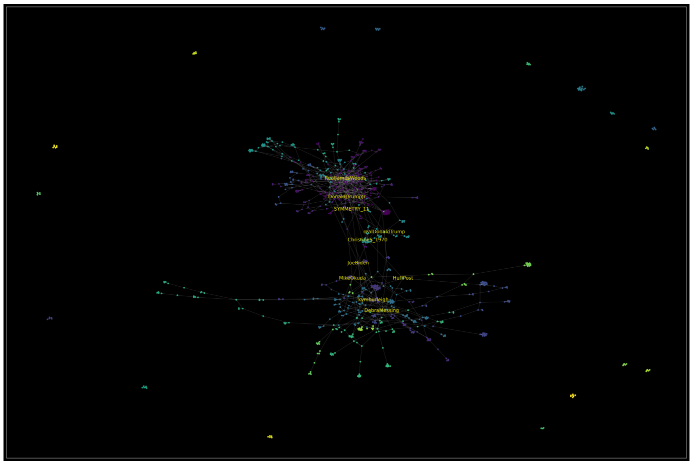

Figure 4. ForceAtlas2 with California

It is really curious to see how one of the most important hub for Donald Trump community is James Woods. He has stated that he was a member of the Democratic Party

[3] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4051631/

until the impeachment of Bill Clinton, commenting that "every single Democrat without exception stood behind a convicted perjurer. That was the end." Woods was a registered Independent during the presidencies of George W. Bush and Barack Obama; he has since joined the Republican Party. [4]
On the other hand, it is interesting to note that Donald Trump's niece is present in the Texas community of Biden: MaryLTrump.
She supported Hillary Clinton during the 2016 presidential election. On July 15, 2020, she said in an ABC News interview conducted by George Stephanopoulos that Donald Trump should resign as president. Mary said that he is "utterly incapable of leading this country, and it's dangerous to allow him to do so." In a July 22, 2020 interview on The Late Show with Stephen Colbert, when asked for her professional opinions, Mary stated that Donald Trump exhibits sociopathic tendencies but not at a high-functioning level like his father, was institutionally insulated from responsibilities, and is never held accountable for his actions. [5]

AMERICAN ANALYSIS

Thanks to gephi we were also able to create three partitions for American graph. This included about 90,000 nodes. Colors are based on modularity and the partitions belong respectively to:

Name	Color
Donald Trump	Pink
JoeBiden	Green
Kamala Harris	Blue

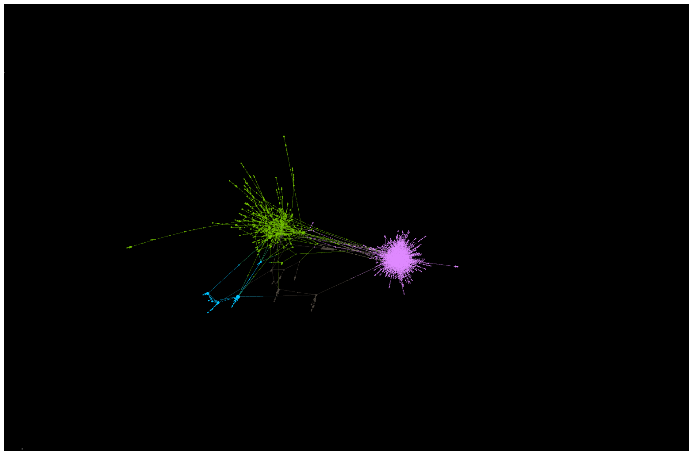

Figure 5. ForceAtlas2 with USA

VI. Data Visualization

In order to show our results, we had to use data visualization tools with dynamic navigation: the purpose is to give the user who wanted to explore the results more choice (for example choosing 'Trump' or 'Biden' subject, or Afinn score or TextBlob polarity) . For this reasons all the results was exported in a Tableau story, that include both sentiment analysis / SNA and some description on the used techniques. Is also available a *Gephi*'s file to show a more detailed information about nodes and hubs in social network analysis.

Conclusion

Jumping to conclusions, we can say that we have found interest results both in sentiment analysis and social network analysis.
In the first study, we observed that in some cases sentiment corresponds with electoral result (as we can see in Texas with Trump), but in other cases happen the opposite situation (Alaska with Trump). In SNA we find two different clusters, concerning to one or the other. Obviously we could have gotten a better result with a large amount of corrected data, but in any case we managed to show interesting facts and behaviours.

[4] https://en.wikipedia.org/wiki/James_Woods
[5] https://en.wikipedia.org/wiki/Mary_L._Trump