Data W
Tweets Sentiment about Donald Trump

Team member:

Jiayuan Hu: jhu125@ucsc.edu

Tianchao Wang: twang91@ucsc.edu

Introduction

Our project is to analyze people's sentiment change toward Donald Trump before and after March 4. On March 4 2017, Trump claimed Obama wiretapped Trump Tower during the election. Donald Trump has issued a variety of executive orders shortly after taking office. So what people evaluate him within the recent months? Did the view change rapidly after this event? We are trying to answer this question by collecting and analyzing tweets We will focus on the results before and after his election with a time horizon of days.
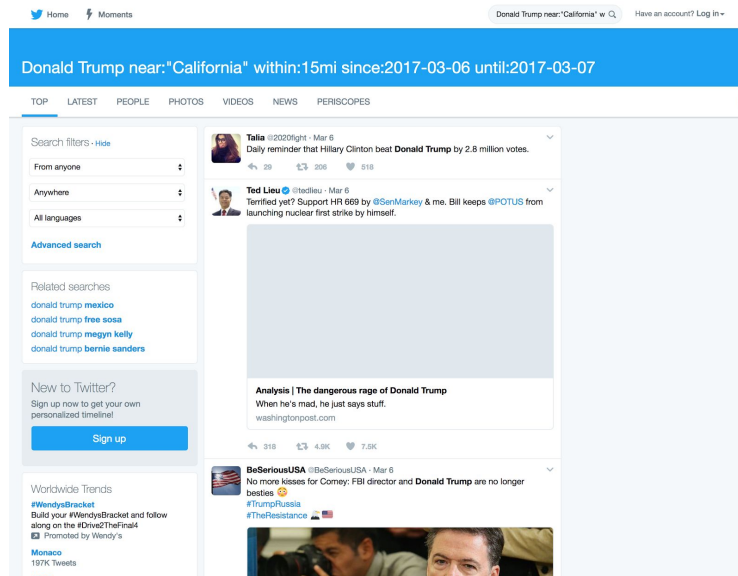
Data Scraping

We found that it is really slow to use Twitter API to get data from Twitter so we use another method to crawl data by constructing a Twitter Advanced Search URL.
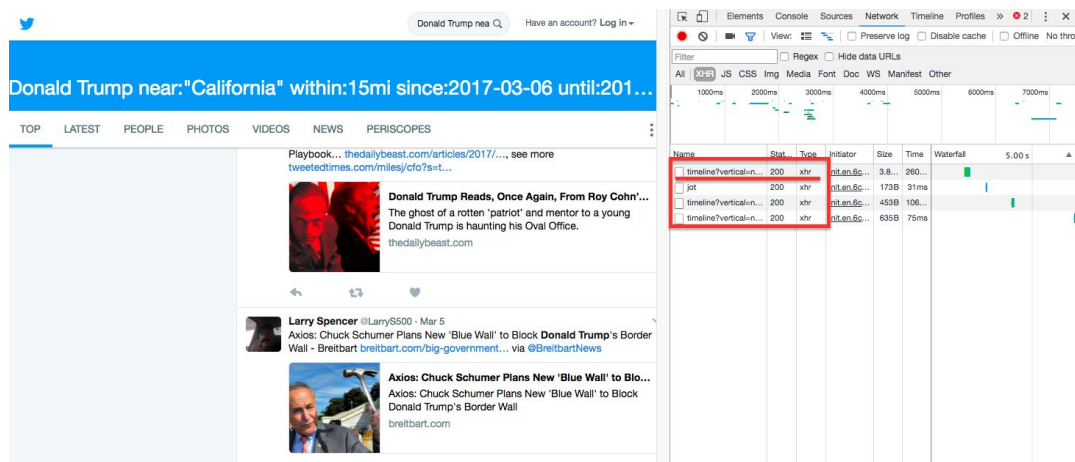
When we begin a Twitter Advanced Search, we use "Donald Trump" and a state name and a date as three keywords, e.g.: "Donald Trump" + "CALIFORNIA" + " 2017-03-06 to 2017-03-07"

When we start search, we can get result like this:

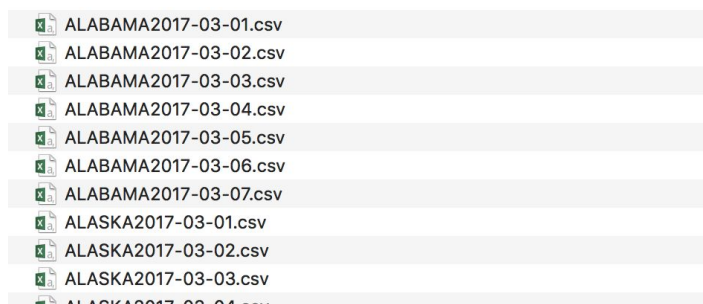Then we can inspect the webpage (we use the Chrome browser):

You may notice that every time we scroll the webpage to the bottom, the webpage generate a link like:

"[https://twitter.com/i/search/timeline?vertical=news&q=Donald%20Trump%20near%3A%22California%22%20within%3A15mi%20since%3A2017-03-06%20until%3A2017-03-07&src=typd&include_available_features=1&include_entities=1&lang=en&max_position=TWEET-838634670162051072-838882663909842944-BD1UO2FFu9QAAAAAAAETAAAAcAAAASAAAIAADAAAAAAABABAAAAAAAAACAAgAAgAAAAAAAAAgAAEAAgAABAAAACggAAAAAQAAAAACAAAAIAAAAAAAAABAAgAAAAACAEQAAAAAAAABAACAAAAAAAAAQAAAAAAAAAAAAAAQAAgAEAABAAAAAAAAAAAAAAAAgABAIAACAAAAAAAAAAA&reset_error_state=false](https://twitter.com/i/search/timeline)"

This link actually refers to a *.json* file which contains the information of the webpage and we can get tweets by reading and parsing the *.json* file. Then we can construct a URL like which shown above.

Our script of web crawler is *"tweets_crawler.py"*. To run this script, you need to install and import some necessary libraries: urllib2, csv, json, datetime, bs4, etc. The script crawl Tweets of 50 states from 2017-03-01 to 2017-03-07. And all this data will be stored in *.csv* files and each file contains no more than 120 tweets. These are raw data for our projects.

ALABAMA2017-03-01.csv
ALABAMA2017-03-02.csv
ALABAMA2017-03-03.csv
ALABAMA2017-03-04.csv
ALABAMA2017-03-05.csv
ALABAMA2017-03-06.csv
ALABAMA2017-03-07.csv
ALASKA2017-03-01.csv
ALASKA2017-03-02.csv
ALASKA2017-03-03.csv
ALASKA2017-03-04.csv

Data Analysis

To get people's sentiment, we use NLTK(Natural Language Toolkit) to analyze every tweet content. And the Sentiment Analyzer in the NLTk will return a sequence of probabilities like: {neg:0.3, pos:0.5, neu:0.2}.

Our Script of tweets sentiment analysis is *"sentiment_analyzer.py"* . To run this script, you may need to install *NLTK* and *NLTK Data: https://www.nltk.org*

Data Visualization

- Data aggregation and word cloud generation

Till now we have got these csv files, and the next step is performing data cleaning and aggregation over them.

Then in script/word_cloud.py, we first imported csv to read these files and created two lists, sentiments and contents, to store the tweets contents and sentiments at each state. Next, we used pandas library to group these contents' words by their occurrence, in this step, we created a text file called stopwords.txt to filter these words who contribute nothing to our analysis, such as he, him, we, then etc.

Finally we used imported a python library called wordcloud, with the help of this library, we can draw a word cloud like this:

350	DonaldTrump	22
1976	status	22
2163	youtu	21
1417	fb	19
818	Russia	18
1955	speech	17
329	Democrats	14
1319	dlvr	14
1113	attempt	14
1885	runs	14
2057	trouble	13
1376	errors	12
587	Like	12

If a word has more occurrence than others, it will appear larger in our final output word cloud image, we can see from the example that in this state's tweets content, women are mentioned more than other words.

- Sentiments Visualization

In the former step, sentiments are dumped as sentiments.js with a json format. In this part, at us_map.js, we read the sentiments files and draw a dynamic webpage to show the word cloud at each state, also, we showed the sentiments at each state by color. If people's views are more negative towards Trump at a state, this state will appear more darker in the final output.

In addition, we add two buttons called before and after, with the help of Ajax calls, we can compare people's views by this event.

The main visualization file is us_map.html. d3.js and bootstrap responsive design are useful in implementing these features.
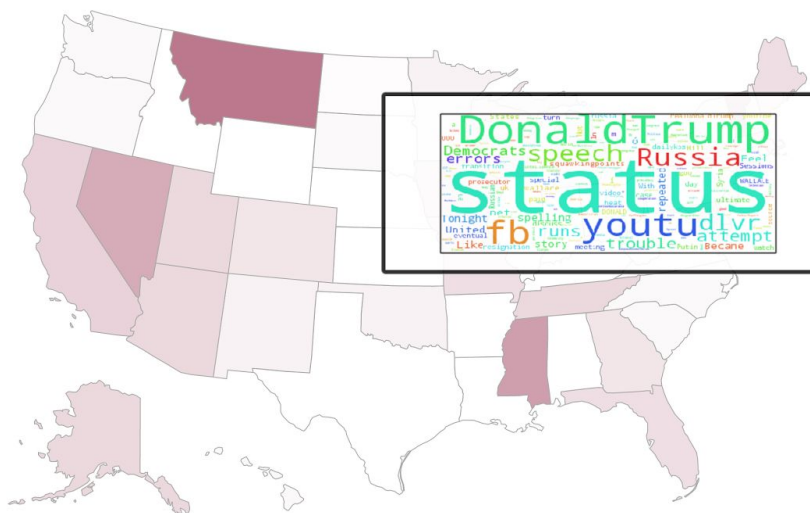
Before March 4:

After March 4:
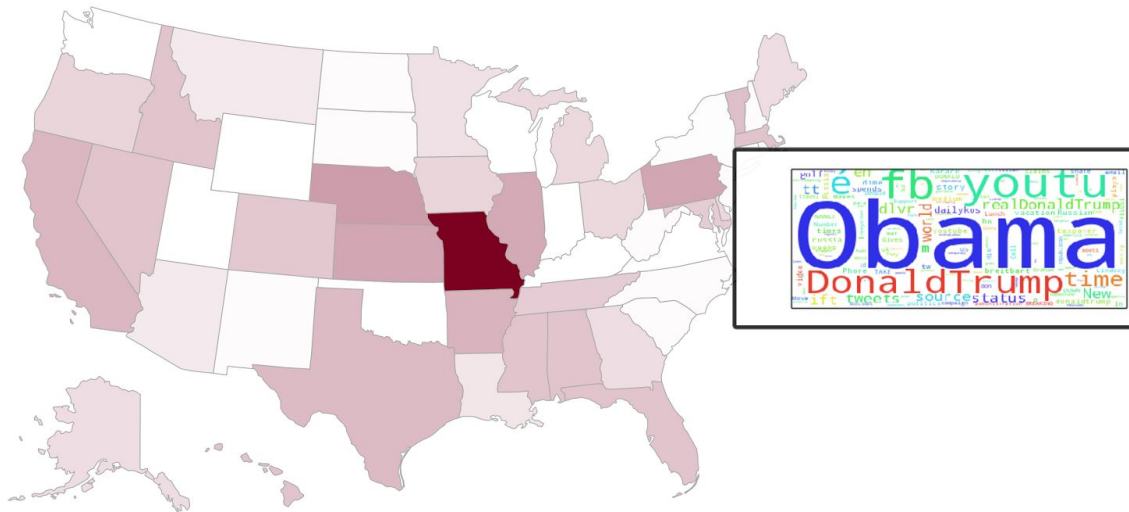
Conclusion

Clearly we can see from above, at most states (especially those who is close to coasts),

the color becomes darker after this event.

Reference
[1] McKinney, W. (2014). *Python for data analysis.* Beijing: O'Reilly.
[2] Tom Dickinson's Blog:
http://tomkdickinson.co.uk/2015/01/scraping-tweets-directly-from-twitters-search-page-part-1/
[3] D3.js: https://d3js.org
[4] Natural Language Toolkit: https://www.nltk.org