The Data

Lobbyists4America is a company that seeks to provide insights to their customers. Their customers aim to affect legislation within the US. The client wants to analyze the 2008-2017 congressional tweets in order to understand

a) Key topics
b) Key members
c) Relationships within Congress.

These insights will help them focus and strengthen their lobbying efforts.

I chose this client and dataset to gain understanding of social media interactions and to gain experience in getting insights from social media data – in this case Twitter.

Importing the data

The data is in the form of 2 JSONs

1) Users.json : list of all relevant twitter users
2) Tweets.json : list of all tweets from the above users

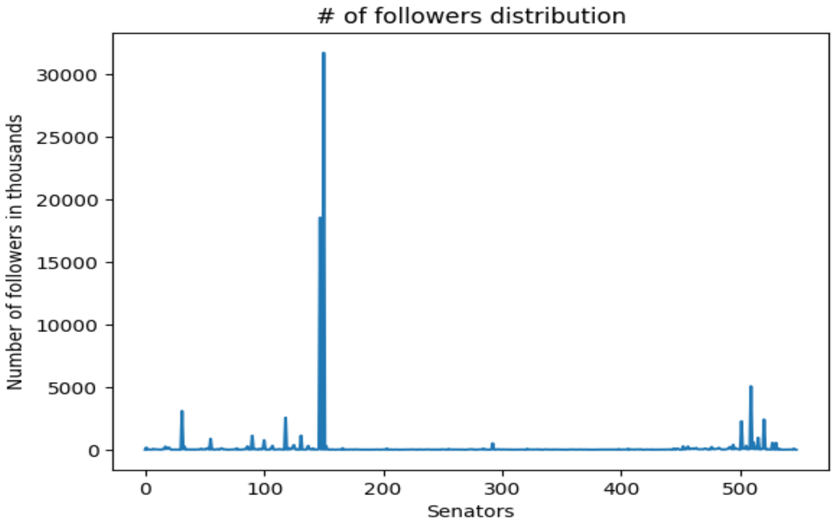
The .json files were read into Python as pandas objects using read_json with a chunksize defined. The pandas dataframes were then read back to a csv format for easier load in the future.

Cleaning the data

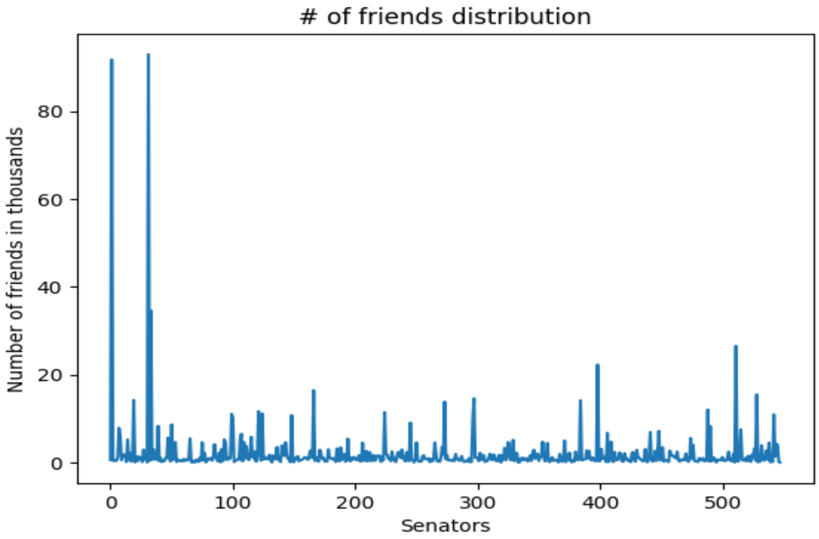
- Checked for missing values in both tables and dropped the following columns (all missing data/ non valuable data/duplicate info)
    - Tweets – contributors,coordinates,geo,withheld_copyright, withheld_in_countries,withheld_scope
- New features created from existing columns
    - Users : Time since user created, tweets per day
    - Tweets : created_date, created_time, created_year
- Changed columns type to extract information
    - Tweets – 'source' from html layout to string using BeautifulSoup, extracted , hashtags used, user ids mentioned in tweets from dict type column
- Removed rows in tweets data not corresponding to users in users table (around 2k)

Understanding the data

A) Users data
    - There are 548 senators that are part of this analysis. On average they have 17k followers.

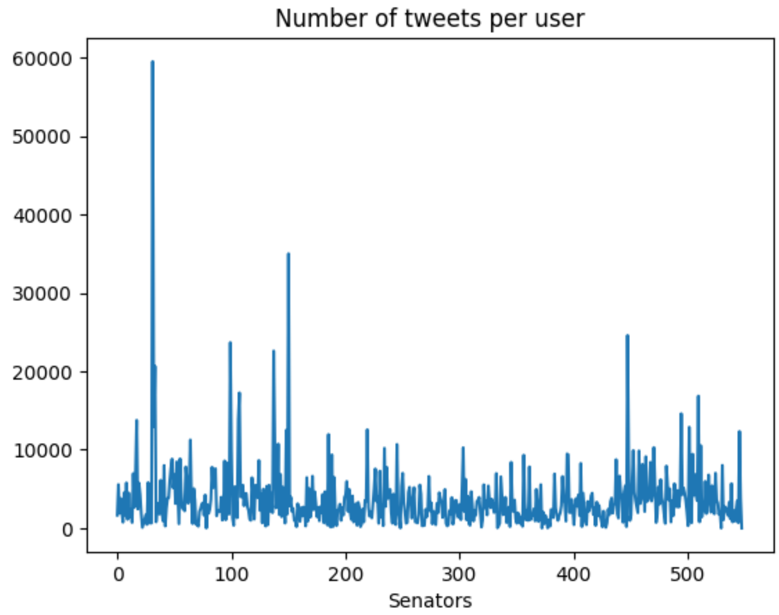


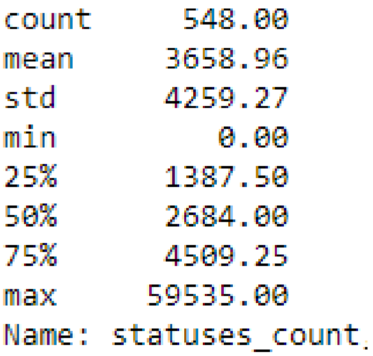
```
count         548.00
mean       163433.91
std       1597357.02
min             4.00
25%          8960.25
50%         16732.00
75%         33081.00
max      31712585.00
Name: followers_count
```



```
count      548.00
mean      2033.73
std       6278.44
min          0.00
25%        368.00
50%        751.50
75%       1670.50
max      92934.00
Name: friends_count
```

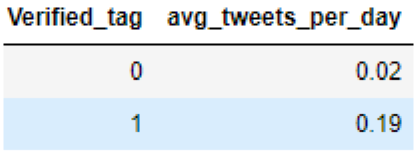
- On average a senator follows 2k accounts but that is highly skewed as third quartile is 1700. A few of these senators follow a large number of accounts. There is no correlation between how many they follow vs how many follow them (friends)





```
count      548.00
mean      3658.96
std       4259.27
min          0.00
25%       1387.50
50%       2684.00
75%       4509.25
max      59535.00
Name: statuses_count
```

- On avg 3.7k tweets. Avg is 0.18 tweets per day.

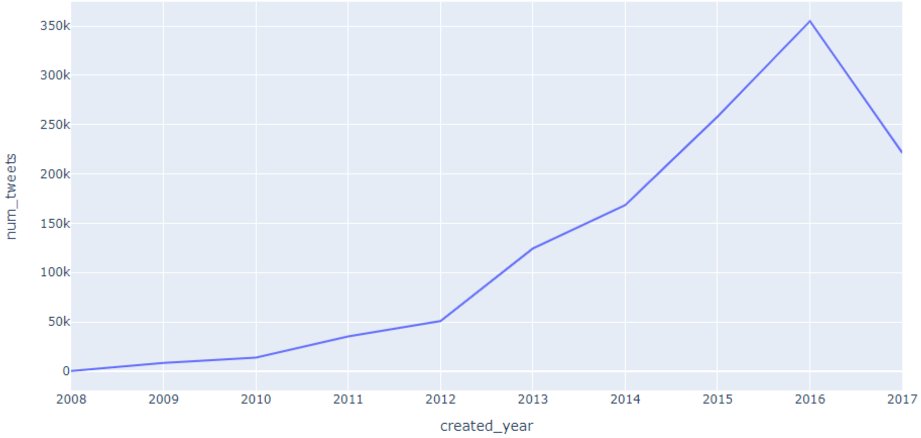


Verified_tag	avg_tweets_per_day
0	0.02
1	0.19

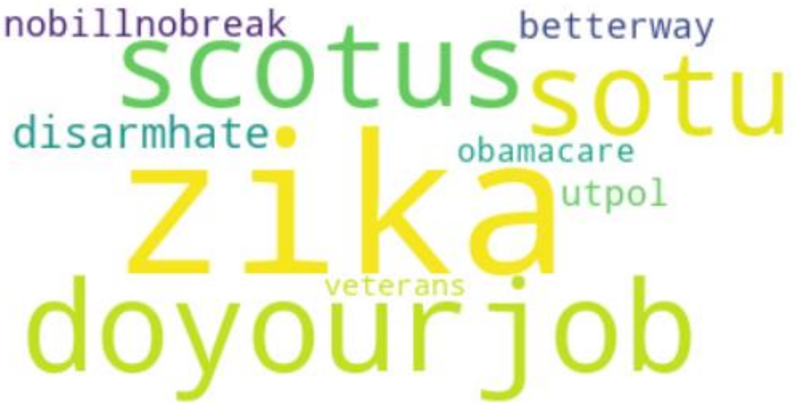
- Verified users tweet much more than non-verified ones
- There is no important data missing in this file.

B) <u>Tweets data</u>

- 1235383 tweets with 21 attributes
- The tweets are from 2008-08-04 to 2017-06-06 – approx. 9 years. There are 3 Presidential election years in it as well – 2008, 2012 and 2016.



- The number of tweets from senators in grown very fast reaching its peak in 2016 with approx. 350k tweets.
- The most tweets in this year (2016) were from John Kasich and Donald Trump. Donald Trump and Bernie Sanders were the top retweeted senators.



- The main topics in 2016 discussed on Twitter by them were about the Zika virus outbreak, the supreme court judge nomination and Republicans "Do your Job" campaign. It was also a presidential re-election year.

<u>Hypothesis</u>

A) Presidential election years see an increase in senator retweeting activity (retweeting and quoting as % of tweets) due to party campaigns
B) There is a segmentation in users based on their interactions (quote/retweet) for 2016
C) Users have specific roles in amplifying these campaigns
D) There are some key influencers within the users

<u>Approach</u>

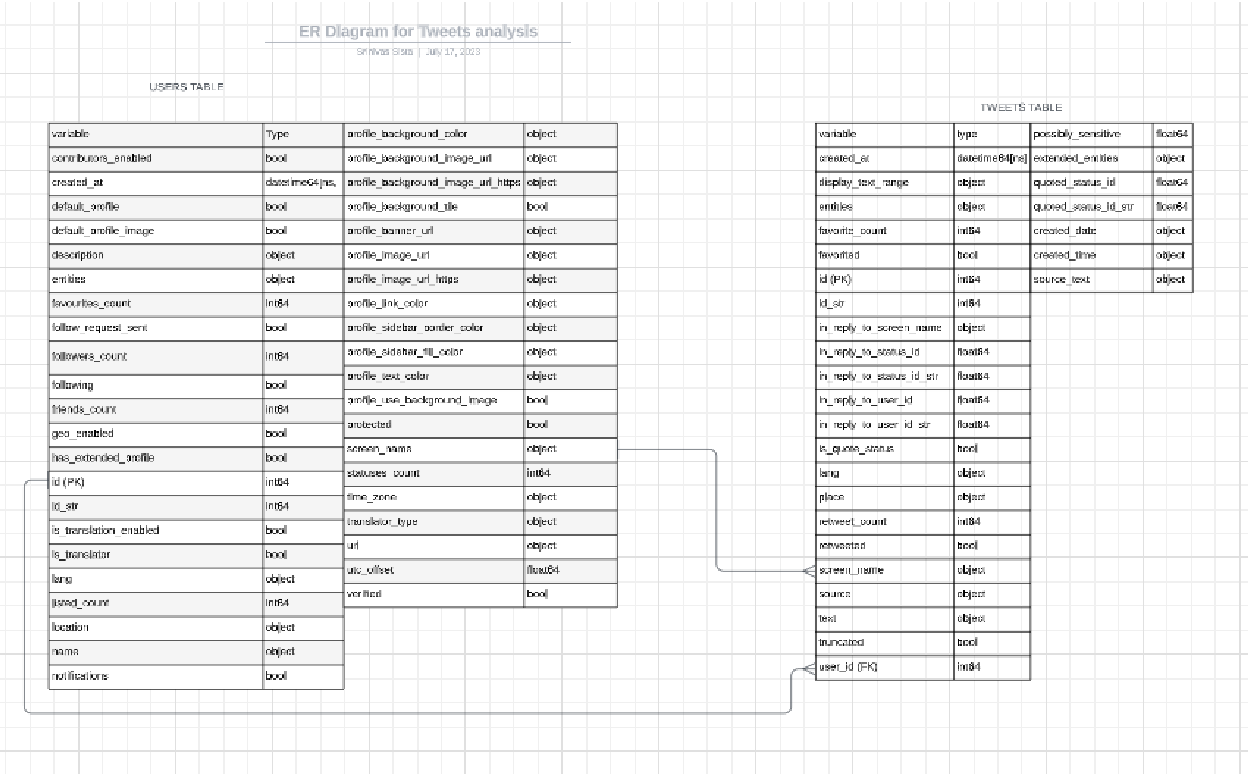
Hypothesis A : Create a new column in users table with % retweets/tweet and check the avg of this ratio over the years.

Hypothesis B: Create a graph where users are nodes and retweeting/quoting is a relationship. Visualise this to identify different segments for 2016 tweets data

Hypothesis C: Segment nodes based on their # of tweets, % of retweets, following to define practical roles within the network ex. Original poster, Amplifier, Not active etc

Hypothesis D: Look for outliers from C to find Top Influencers

<u>ER Diagram</u>



ER Diagram for Tweets analysis

Srinivas Sista | July 17, 2023

USERS TABLE

variable	Type	profile_background_color	object
contributors_enabled	bool	profile_background_image_url	object
created_at	datetime64[ns,	profile_background_image_url_https	object
default_profile	bool	profile_background_tile	bool
default_profile_image	bool	profile_banner_url	object
description	object	profile_image_url	object
entities	object	profile_image_url_https	object
favourites_count	int64	profile_link_color	object
follow_request_sent	bool	profile_sidebar_border_color	object
followers_count	int64	profile_sidebar_fill_color	object
following	bool	profile_text_color	object
friends_count	int64	profile_use_background_image	bool
geo_enabled	bool	protected	bool
has_extended_profile	bool	screen_name	object
id (PK)	int64	statuses_count	int64
id_str	int64	time_zone	object
is_translation_enabled	bool	translator_type	object
is_translator	bool	url	object
lang	object	utc_offset	float64
listed_count	int64	verified	bool
location	object		
name	object		
notifications	bool		

TWEETS TABLE

variable	type	possibly_sensitive	float64
created_at	datetime64[ns]	extended_entities	object
display_text_range	object	quoted_status_id	float64
entities	object	quoted_status_id_str	float64
favorite_count	int64	created_date	object
favorited	bool	created_time	object
id (PK)	int64	source_text	object
id_str	int64		
in_reply_to_screen_name	object		
in_reply_to_status_id	float64		
in_reply_to_status_id_str	float64		
in_reply_to_user_id	float64		
in_reply_to_user_id_str	float64		
is_quote_status	bool		
lang	object		
place	object		
retweet_count	int64		
retweeted	bool		
screen_name	object		
source	object		
text	object		
truncated	bool		
user_id (FK)	int64		