Lobbyists4America dataset

Filip Vranješević

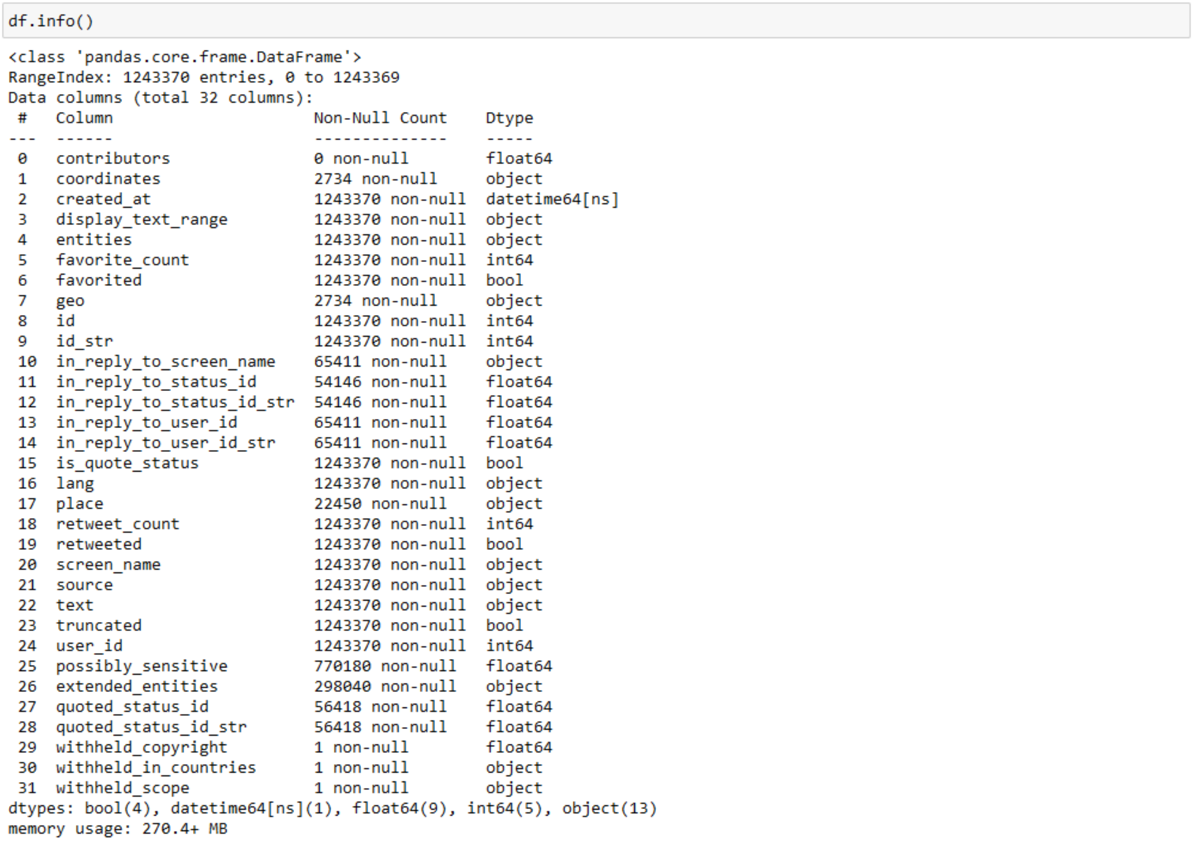
1. Preparing a dataset

1.1. Which dataset and why?

I chose to import the Lobbyists4America dataset for the lobbying company. The area of online electioneering has exploded in the past ten years, with many data analysis companies providing targeted ad services to influence voters. Many such companies have been at the centre of controversy, such as Cambridge Analytica, a company that specialized in using "big data and advanced psychographics" in electoral campaigns to target voters using data provided by Facebook. Likewise, targeted lobbying of politicians using a data analysis approach might prove to be increasingly useful as politicians increasingly interact online with each other as well as with their constituents. One might envisage using social media data analysis to pick out politicians whose opinions are most malleable on a particular issue (e.g. healthcare reform) as well as the best way to get in communication with them (are they likely to respond on Twitter? which of their colleagues do they follow most closely?).

1.2. Importing the Lobbyists4America dataset

The Lobbyists4America dataset is a gzip file which, when unzipped. gives two files: *tweets* and *users*. Reading the header makes it clear that both are json files. The *tweets* file is much larger (1243370 entries) and had to be read in using the read_json *chunksize* parameter. The chunks were then concatenated using pd.concat.



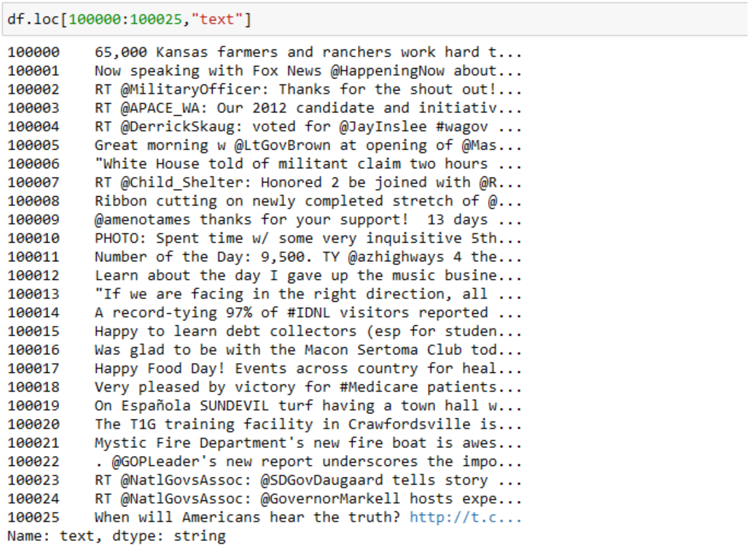
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1243370 entries, 0 to 1243369
Data columns (total 32 columns):
 #   Column                     Non-Null Count    Dtype
---  ------                     --------------    -----
 0   contributors               0 non-null        float64
 1   coordinates                2734 non-null     object
 2   created_at                 1243370 non-null  datetime64[ns]
 3   display_text_range         1243370 non-null  object
 4   entities                   1243370 non-null  object
 5   favorite_count             1243370 non-null  int64
 6   favorited                  1243370 non-null  bool
 7   geo                        2734 non-null     object
 8   id                         1243370 non-null  int64
 9   id_str                     1243370 non-null  int64
 10  in_reply_to_screen_name    65411 non-null    object
 11  in_reply_to_status_id      54146 non-null    float64
 12  in_reply_to_status_id_str  54146 non-null    float64
 13  in_reply_to_user_id        65411 non-null    float64
 14  in_reply_to_user_id_str    65411 non-null    float64
 15  is_quote_status            1243370 non-null  bool
 16  lang                       1243370 non-null  object
 17  place                      22450 non-null    object
 18  retweet_count              1243370 non-null  int64
 19  retweeted                  1243370 non-null  bool
 20  screen_name                1243370 non-null  object
 21  source                     1243370 non-null  object
 22  text                       1243370 non-null  object
 23  truncated                  1243370 non-null  bool
 24  user_id                    1243370 non-null  int64
 25  possibly_sensitive         770180 non-null   float64
 26  extended_entities          298040 non-null   object
 27  quoted_status_id           56418 non-null    float64
 28  quoted_status_id_str       56418 non-null    float64
 29  withheld_copyright         1 non-null        float64
 30  withheld_in_countries      1 non-null        object
 31  withheld_scope             1 non-null        object
dtypes: bool(4), datetime64[ns](1), float64(9), int64(5), object(13)
memory usage: 270.4+ MB
```

Some columns contain zero non-null values (“contributors”) and the last three have only one (a 2016 tweet by John Kasich claimed by copyright owners) so I removed those columns. Created_at is a datetime type, which is good, but I would also like the text to be a string type with:

df = df.astype({"text":"string"})

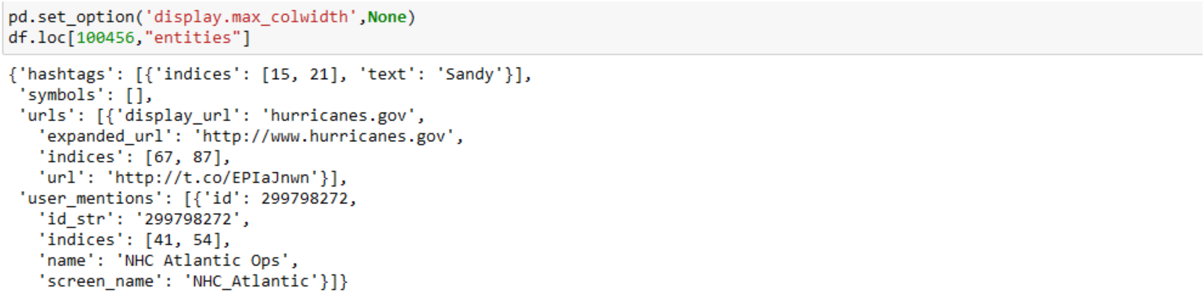
so we can more easily explore the text.



```
df.loc[100000:100025,"text"]
```

```
100000    65,000 Kansas farmers and ranchers work hard t...
100001    Now speaking with Fox News @HappeningNow about...
100002    RT @MilitaryOfficer: Thanks for the shout out!...
100003    RT @APACE_WA: Our 2012 candidate and initiativ...
100004    RT @DerrickSkaug: voted for @JayInslee #wagov ...
100005    Great morning w @LtGovBrown at opening of @Mas...
100006    "White House told of militant claim two hours ...
100007    RT @Child_Shelter: Honored 2 be joined with @R...
100008    Ribbon cutting on newly completed stretch of @...
100009    @amenotames thanks for your support!  13 days ...
100010    PHOTO: Spent time w/ some very inquisitive 5th...
100011    Number of the Day: 9,500. TY @azhighways 4 the...
100012    Learn about the day I gave up the music busine...
100013    "If we are facing in the right direction, all ...
100014    A record-tying 97% of #IDNL visitors reported ...
100015    Happy to learn debt collectors (esp for studen...
100016    Was glad to be with the Macon Sertoma Club tod...
100017    Happy Food Day! Events across country for heal...
100018    Very pleased by victory for #Medicare patients...
100019    On Española SUNDEVIL turf having a town hall w...
100020    The T1G training facility in Crawfordsville is...
100021    Mystic Fire Department's new fire boat is awes...
100022    . @GOPLeader's new report underscores the impo...
100023    RT @NatlGovsAssoc: @SDGovDaugaard tells story ...
100024    RT @NatlGovsAssoc: @GovernorMarkell hosts expe...
100025    When will Americans hear the truth? http://t.c...
Name: text, dtype: string
```

The column “entities” contains a dictionary with the keys “hashtags”, ”symbols”,”urls” and “user_mentions”, which will be very useful to turn into a DataFrame later and manipulate further.

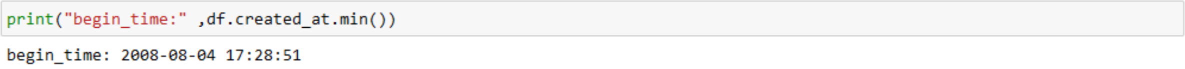


```
pd.set_option('display.max_colwidth',None)
df.loc[100456,"entities"]
```

```
{'hashtags': [{'indices': [15, 21], 'text': 'Sandy'}],
 'symbols': [],
 'urls': [{'display_url': 'hurricanes.gov',
   'expanded_url': 'http://www.hurricanes.gov',
   'indices': [67, 87],
   'url': 'http://t.co/EPIaJnwn'}],
 'user_mentions': [{'id': 299798272,
   'id_str': '299798272',
   'indices': [41, 54],
   'name': 'NHC Atlantic Ops',
   'screen_name': 'NHC_Atlantic'}]}
```

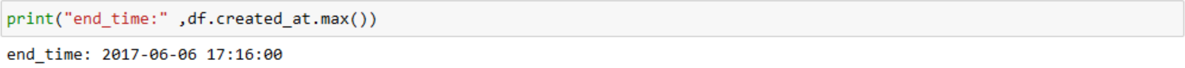
1.3. Initial exploration of the dataset

Finally, we can look at some basic facts about our datased. Firstly, we can determine the time boundaries of our dataset:



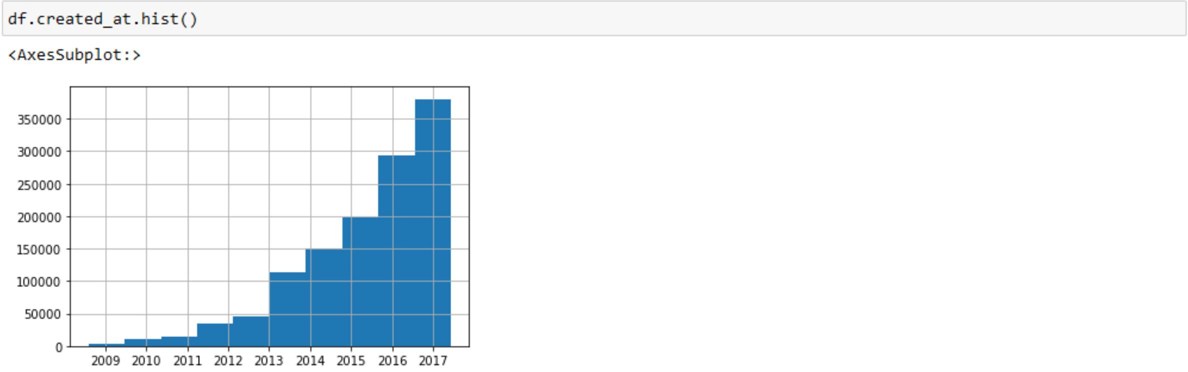
```
print("begin_time:" ,df.created_at.min())
```

```
begin_time: 2008-08-04 17:28:51
```



```
print("end_time:" ,df.created_at.max())
```

```
end_time: 2017-06-06 17:16:00
```

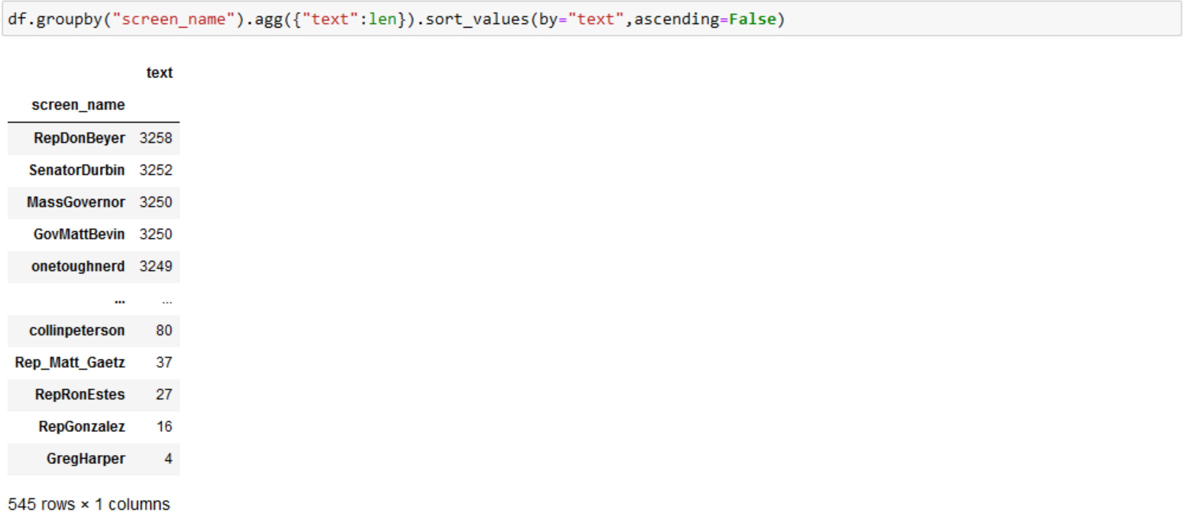


```
df.created_at.hist()
```

```
<AxesSubplot:>
```

The continually increasing popularity of Twitter to communicate political messages from 2008-2017 was to be expected.

Then, we can perform a very basic analysis. We can count the number of distinct users in the tweets table and see whether the number matches the number of entries in the users table. Also ,we can sort the users by the number of tweets to find the most prolific tweeters.

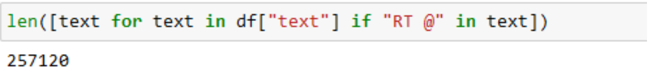


```
df.groupby("screen_name").agg({"text":len}).sort_values(by="text",ascending=False)
```

	text
screen_name	
RepDonBeyer	3258
SenatorDurbin	3252
MassGovernor	3250
GovMattBevin	3250
onetoughnerd	3249
...	...
collinpeterson	80
Rep_Matt_Gaetz	37
RepRonEstes	27
RepGonzalez	16
GregHarper	4

545 rows × 1 columns

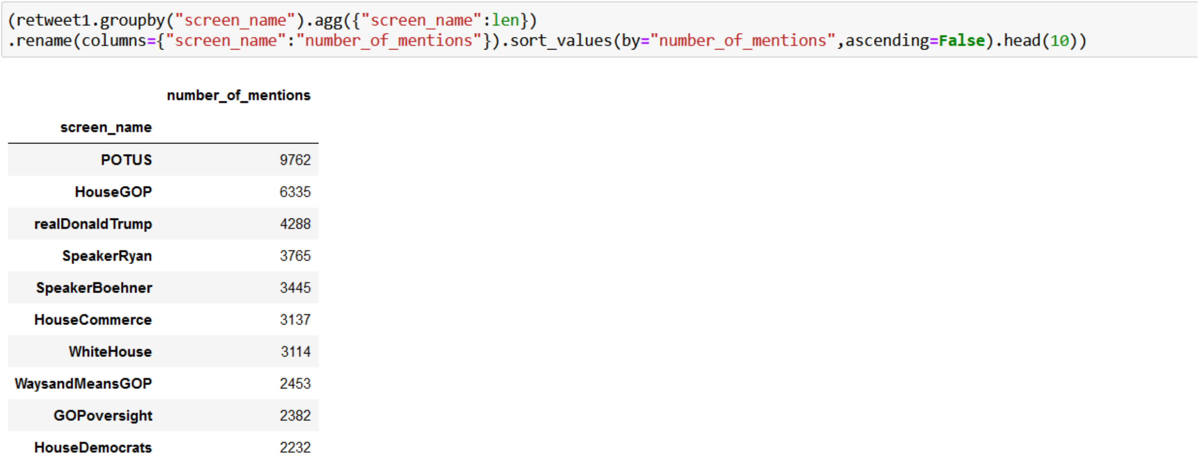
There are 545 rows, compared to 548 entries in the users table, and the greatest number of tweets from a single user is 3258. We can also check how often politicians retweet other users.



```
len([text for text in df["text"] if "RT @" in text])
```

```
257120
```

So, about a fifth of the tweets are retweets. Finally, we can see the most common people mentioned in the tweets using the entity column and doing some grouping and sorting.

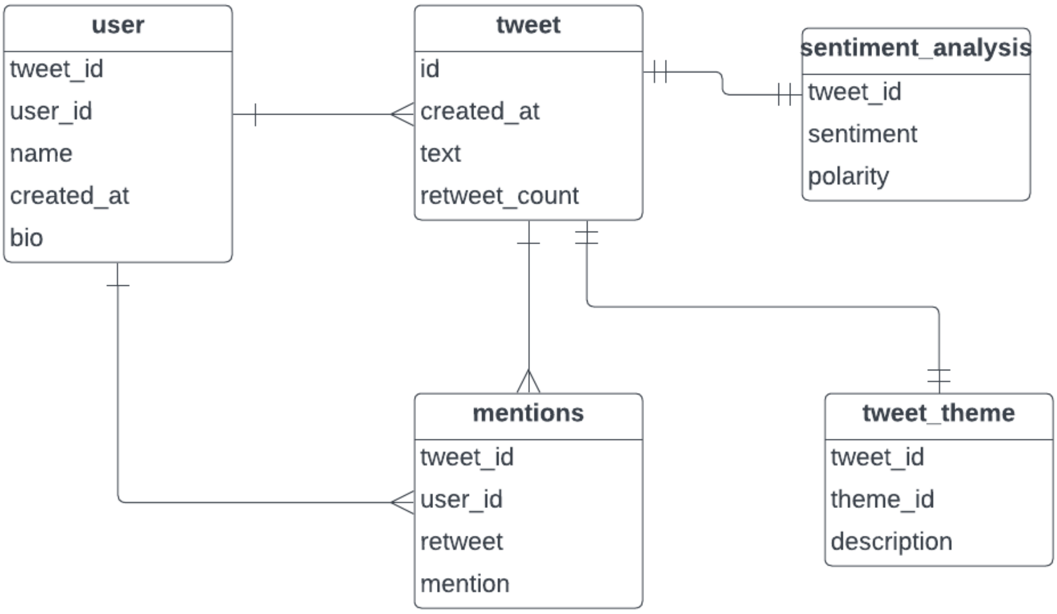


```
(retweet1.groupby("screen_name").agg({"screen_name":len})
.rename(columns={"screen_name":"number_of_mentions"}).sort_values(by="number_of_mentions",ascending=False).head(10))
```

screen_name	number_of_mentions
POTUS	9762
HouseGOP	6335
realDonaldTrump	4288
SpeakerRyan	3765
SpeakerBoehner	3445
HouseCommerce	3137
WhiteHouse	3114
WaysandMeansGOP	2453
GOPoversight	2382
HouseDemocrats	2232

The results are pretty unsurprising.

1.4. ERD diagram



2. Hypotheses

2.1. The aim of the project

This analysis might be interesting to any companies or lobby groups that want to influence congressional/senate votes directly or indirectly. Many politicians interact with other politicians as well as their constituents online and these relationships can be characterized in several different ways (collaborative/oppositional, weak/strong, etc). Exploiting these relationships as well as politicians' tweeting habits can help us gouge how strongly they feel on a certain topic and whether it's worth trying to sway their opinions. We can see the manner of online communication of political figures (do they mostly retweet or post impersonal messages? do they openly argue with opponents?). We can also find which organizations different politicians have relationships with and divide those into partisan and non-partisan relationships. Finally, we can broadly divide politicians by area of interest (healthcare/military/foreign policy/immigration/etc) and see what their attitudes are (positive/negative/ambivalent). In this way, we can build a detailed image of politicians by both topics of interest, types of interactions and online networks. This might be very useful to a person or group seeking the best way to grab a politician's interest and influence their attitudes on a certain topic.

2.2. Questions and hypotheses

Some questions I will attempt to answer:

- Is a politician's popularity on Twitter correlated to the frequency of tweeting? There will be obvious outliers, of course (e.g POTUS).
- Is there correlation between who the politicians engage with (other politicians/ ”ordinary people”/organisations and think-tanks) and other attributes (party, seniority, follower count)?
- Are there significantly different tweeting habits between politicians of different parties?

My hypotheses:

- Politicians that tweet across party lines, i.e. engage directly with their political opponents, are more popular (have a higher follower/retweet count) than politicians that only engage with their own side.
- Politicians engage with more contentious topics (immigration/crime/abortion) in election years (2008/2012/2016) and express more polarized opinions.
- Politicians that express less polarized opinions are more likely to engage with organisations or think tanks than with human users (politicians or otherwise).

2.3. The approach

There are, broadly speaking, two approaches to analysing this data. Firstly, analysing the contents of the tweets to find the topics covered by the tweets as well as the attitudes displayed in the tweets. This can then be analysed with respect to the attributes of the tweeter (party, follower count) as well as with respect to the time (when were certain topics most likely to be mentioned?). In the first case, we can use a statistical analysis to find the correlations between themes, polarities and other attributes. We can visually represent when certain topics are most popular.

The second approach is to look at the interaction networks of politicians. How many politicians/organisations have they retweeted/mentioned? What is the partisan orientation of these organisations? We can perform a t-test to find whether politicians of one major party are more likely to retweet/mention a political opponent, ally or an organisation/think tank.

-