**CRISP-DM**

1. **Project understanding**

The domain: Politics: the 2020 US election.

Who can need the results and how this project can help: the project can help to optimize electoral campaign spending. The political party can use data modelling to decided in which states to invest more money and resources in.

Questions that could be interesting to answer:

* What is the political polarization in the United States?
* Is there a correlation between the number of tweets and external events (debates, local elections…)?
* Сan users be segmented in accordance with their demographic and linguistic description?
* In which countries other than the United States of America do users actively tweet about elections? Which candidate is supported the most in foreign countries?
* Whose tweets are the most often re-tweeted? (find the influencers)
* Is there a difference in attitude towards candidates between English-speaking and Hispanic US citizens, city and rural areas?
* Who is most likely to win the election? (gets the highest support according to tweeter top-10)
* Is there a correlation between the time a user joined Twetter and their declared **political affiliation?**

Main questions we decided to answer (objective):

* What is the political polarization in the United States?
* Is there a difference in attitude towards candidates between English-speaking and Hispanic US citizens?

Deliverable: the model that clusters the US states based on the average sentiment analysis of the tweets geolocalized in the specific state.

Success criteria: after a specific time (i.e. 1 week), if a state that was clustered as pending, and in which we invested money for the political campaign, is now clustered as a favorable state.

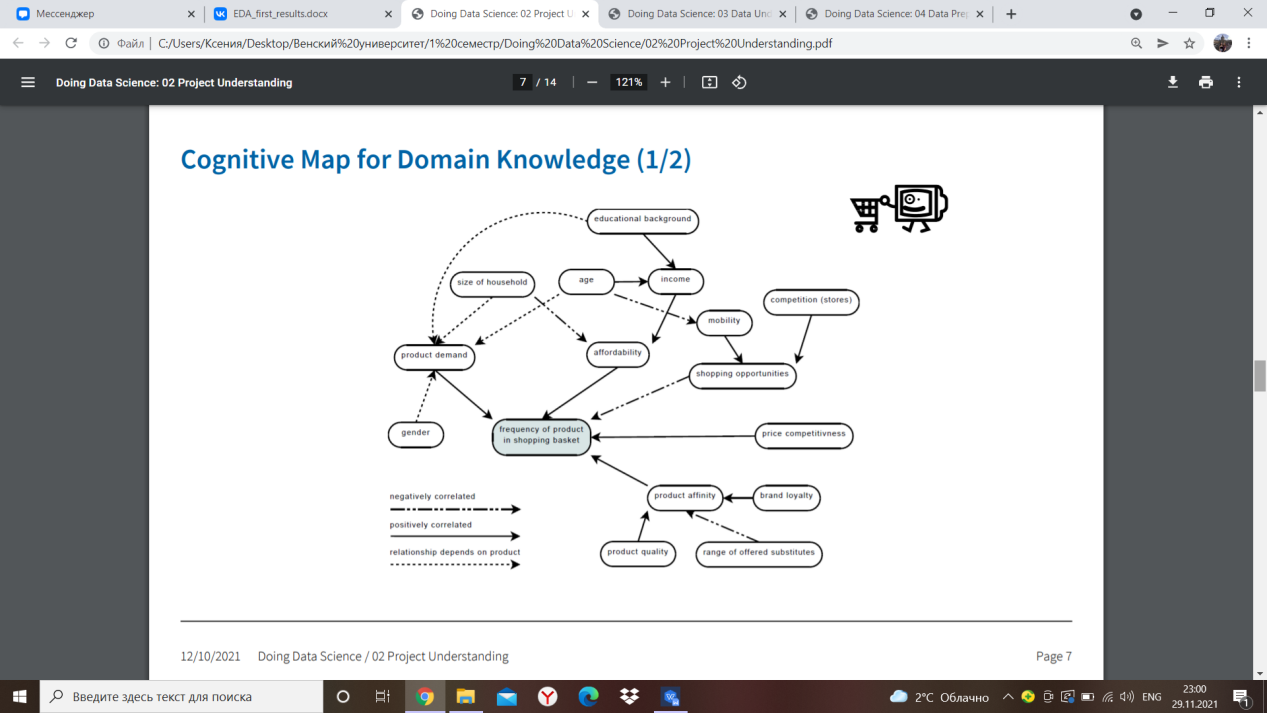
How the solution should look like: the python code with comments on what was done + the report on the gained results in format of posters (video presentation) + graphs and the visualization.

Estimation of chances that the project will be successful: Twitter users can not represent the whole public opinion, because this users are not chosen on the basis of careful sampling by opinion pollsters. According to researches[[1]](#footnote-0) most tweeter users, who take part in political debates, are highly motivated young people, frequently men.

Some users that wrote tweets even in the USA territory can be just tourists (not USA citizens), so may not have any influence on the election results.

So, the chances ~ 70%.

+ WE CAN USE A COGNITIVE MAP??



+ We can visualize timeline of the project (Gantt chart)

1. **Data understanding**

Data structure: tabular format

Data quality: there are many missing values in both data sets. Data is up-to-date and valid.

Types of data: float64(6), object(11) (in both datasets).

Outliers: visible growth in the number of tweets according to external events (day after Last debate, day after elections, day after the results were announced).

Relationships in data:?

**(Manos - EDA)**

Programming language used for Analysis: Python

First, let’s figure out what the dataset consists of:

RangeIndex: 776886 entries, 0 to 776885

Data columns (total 17 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 created\_at 776886 non-null object

1 tweet 776886 non-null object

2 likes 776886 non-null float64

3 retweet\_count 776886 non-null float64

4 user\_id 776886 non-null float64

5 user\_name 776868 non-null object

6 user\_screen\_name 776886 non-null object

7 user\_join\_date 776886 non-null object

8 user\_followers\_count 776886 non-null float64

9 user\_location 543095 non-null object

10 lat 355293 non-null float64

11 long 355293 non-null float64

12 city 186872 non-null object

13 country 353779 non-null object

14 continent 353797 non-null object

15 state 260195 non-null object

16 state\_code 244609 non-null object

dtypes: float64(6), object(11)

memory usage: 100.8+ MB

This is the pandas Dataframe output for the Biden dataset. Something similar is there for the Trump dataset:

RangeIndex: 970919 entries, 0 to 970918

Data columns (total 17 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 created\_at 970919 non-null object

1 tweet 970919 non-null object

2 likes 970919 non-null float64

3 retweet\_count 970919 non-null float64

4 user\_id 970919 non-null float64

5 user\_name 970903 non-null object

6 user\_screen\_name 970919 non-null object

7 user\_join\_date 970919 non-null object

8 user\_followers\_count 970919 non-null float64

9 user\_location 675966 non-null object

10 lat 445719 non-null float64

11 long 445719 non-null float64

12 city 227187 non-null object

13 country 442748 non-null object

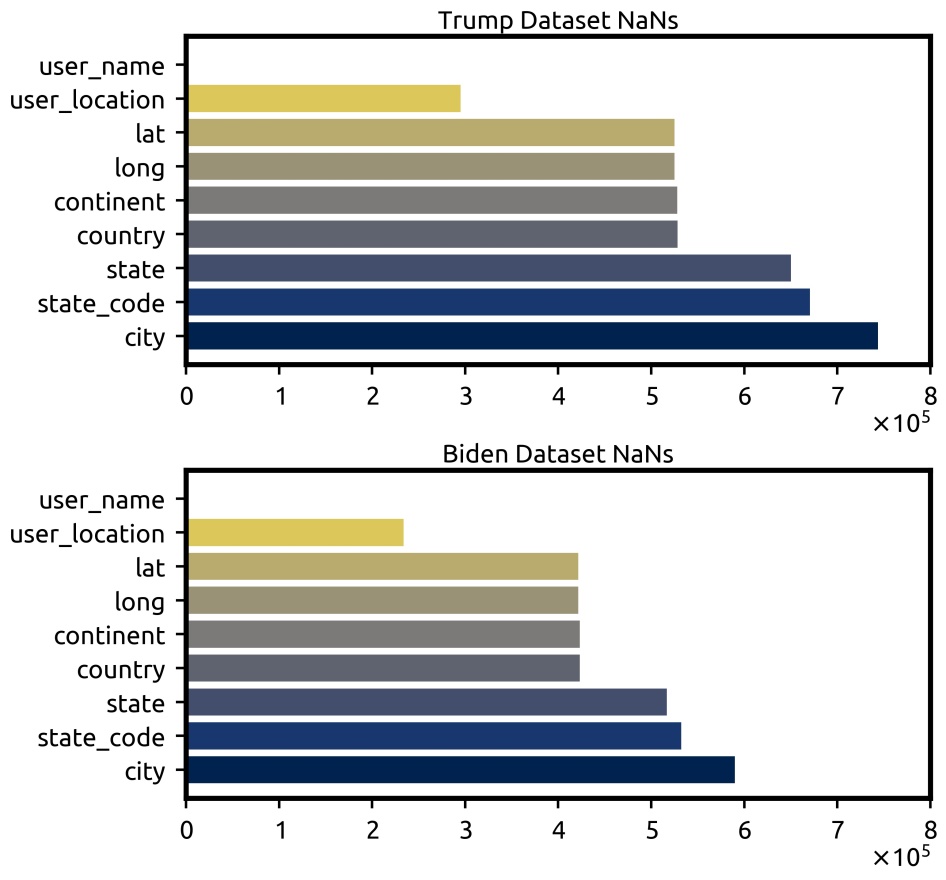
14 continent 442765 non-null object

15 state 320620 non-null object

16 state\_code 300425 non-null object

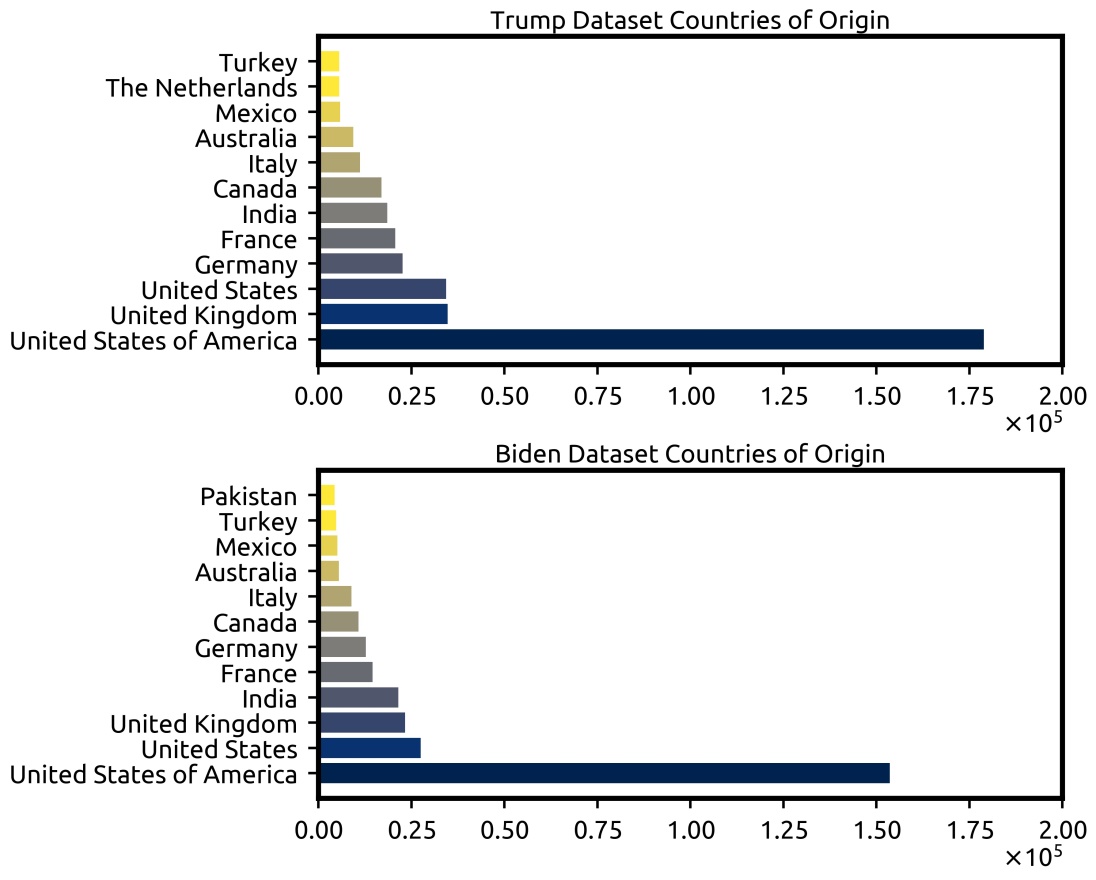
dtypes: float64(6), object(11)

Now let’s see how many NaNs are there per column:

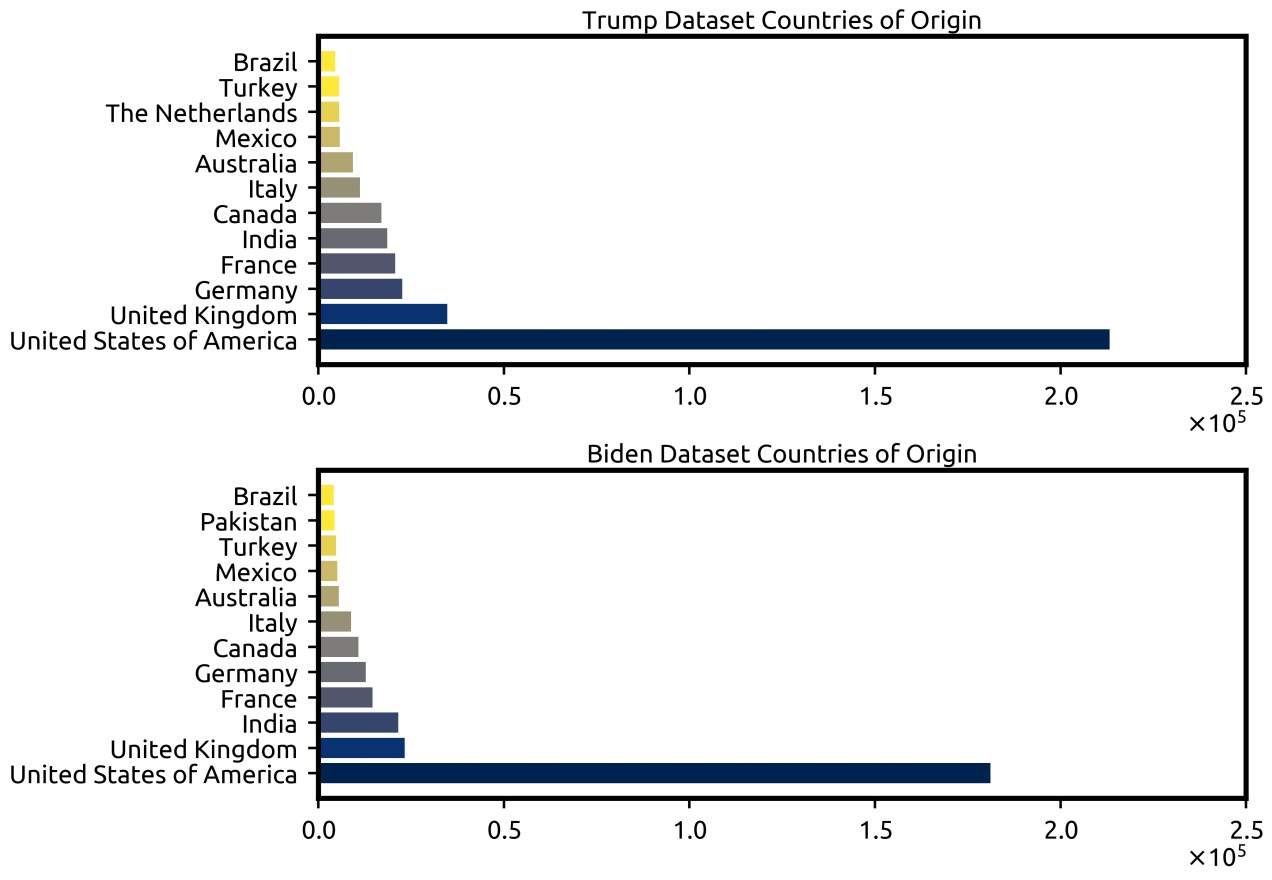


It is clear that in the case of location info (e.g. latitude, longitude, state, country etc.) more than half of the entries are NaNs. Would this dataset then be useful for predicting the location (country or city/countryside) of a generated tweet? For example, by splitting the datasets in parts and using one part to train a model and the second to use it to verify predictions? This becomes even worse if we decide to stick only to people in the United States.

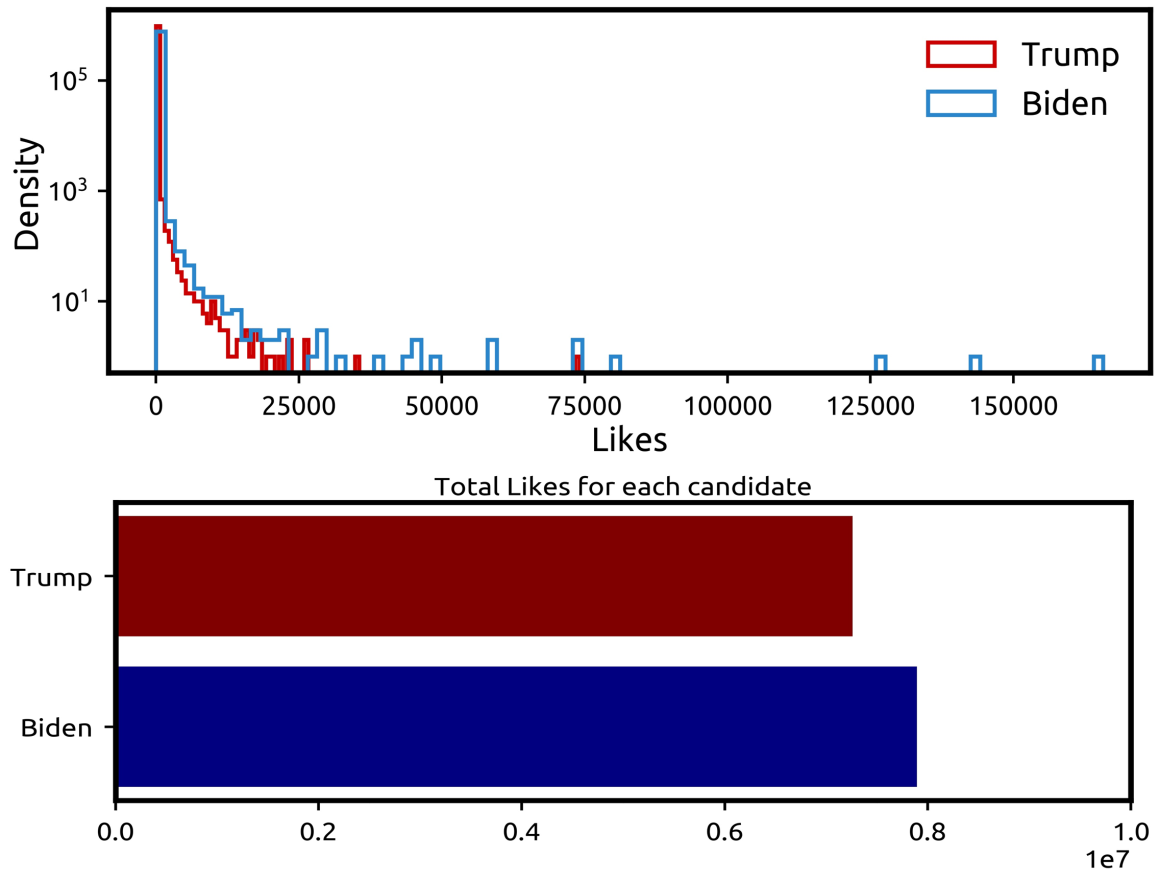
Moreover, there is an issue with the dataset: There are two different entries regarding people in USA, namely ‘United States of America’ and ‘United States’. We find this by plotting 12 countries from which most tweets were generated.



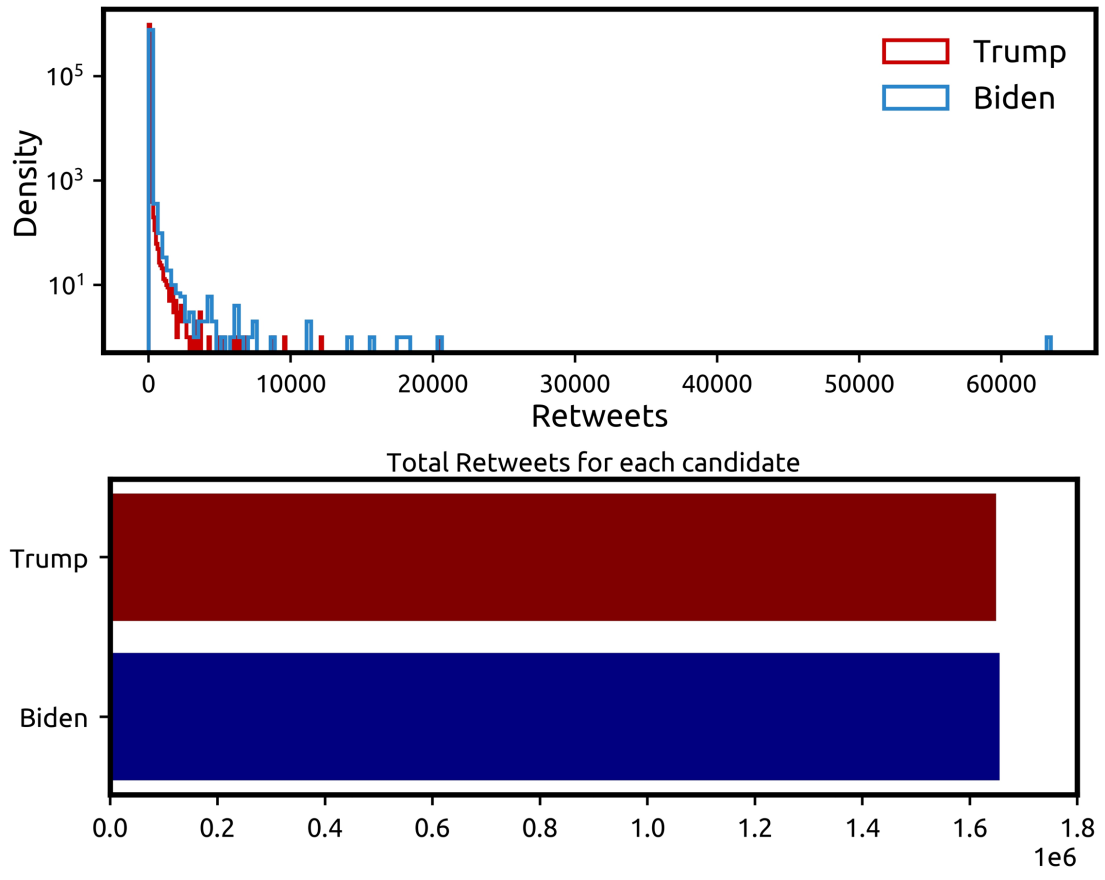
This is how the number of likes look like after unifying entries under ‘United States of America’:



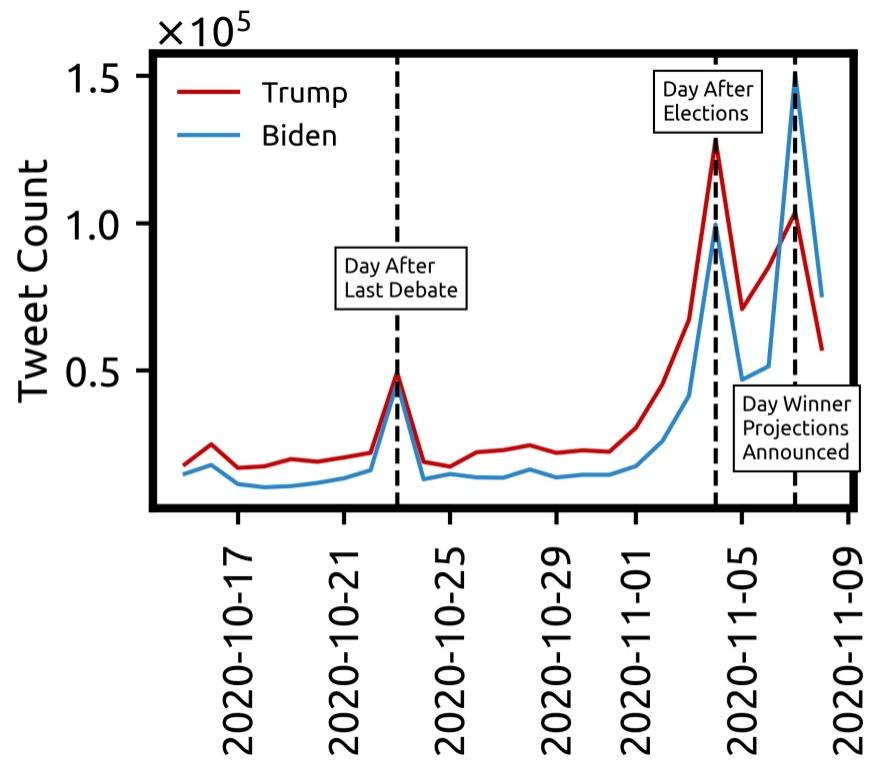
Regarding our other idea, trying to figure out who won the elections based on twitter, below you can see the distribution of likes for the two candidates as well as the number of total likes. It is clear from both plots that more likes were given for Biden than for Trump.



Moreover, below you can see the distribution of retweets for the two candidates as well as the number of total retweets. From the bar plot, it is not quite clear that Biden is ahead, but it is easily distinguished from the histogram plot since the area below the Biden curve is larger than the Trump curve. Furthermore, for both likes and retweets one can find posts for Biden that are extremely popular while this is not the case for Trump (see histograms).

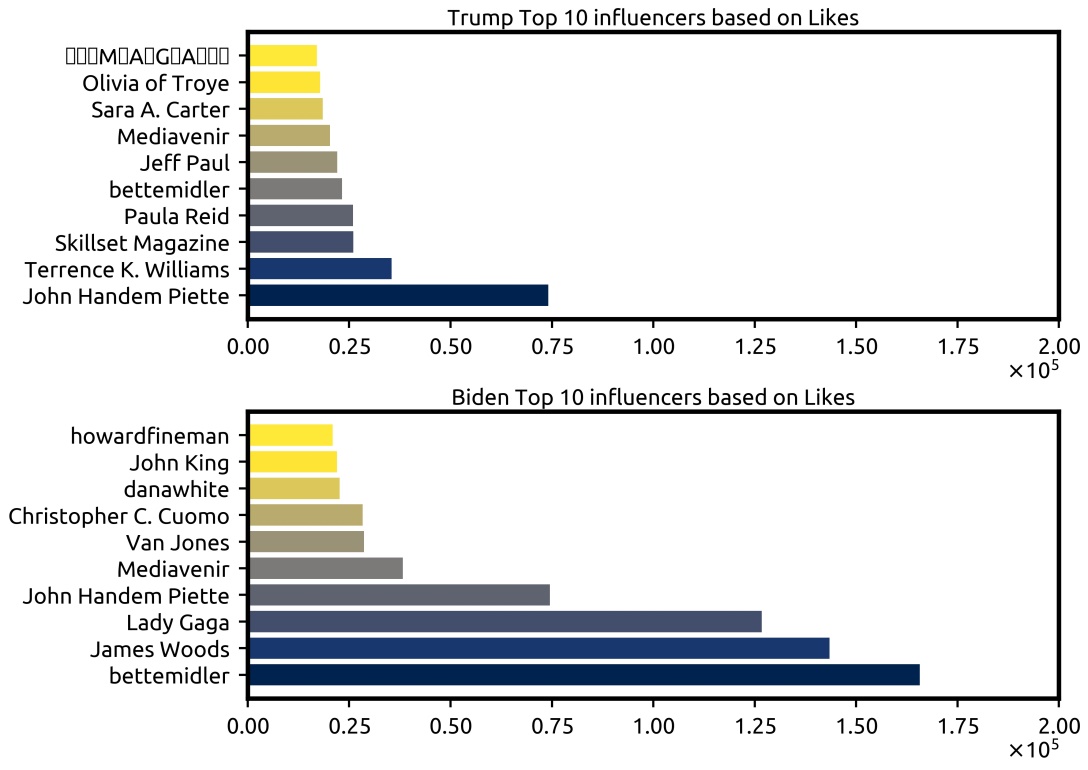


Regarding the time evolution of tweets generated for the two candidates, there are sharp spikes the day after major events such as debates or Election Day, but overall Trump related tweets are more, except the days following the projection from major TV networks that Biden wins the elections.

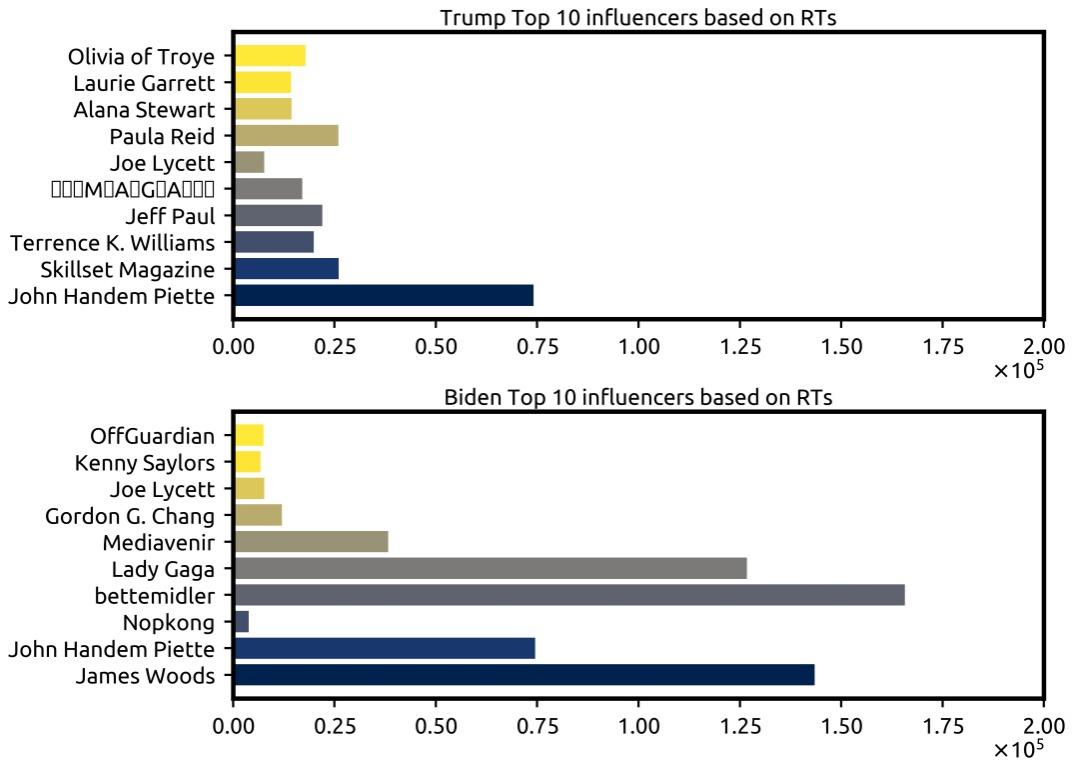


Regarding the idea of ranking influencers

According to Likes:



According to retweets:



1. **Data preparation.**

We need to clean data. Data often need to be manipulated and converted into forms that yield better results.

1. Select attributes and data records.

* We decided to use only US tweets for the clustering (international tweets could bias).
* Among attributes we excluded: tweet\_id: Unique ID of the tweet; source: Utility used to post tweet; user\_id: User ID of tweet creator; user\_name: Username of tweet creator; user\_screen\_name: Screen name of tweet creator; user\_join\_date: Join date of tweet creator; collected\_at: Date and time tweet data was mined from twitter.

1. Missing values

* We excluded NaN values

1. Outliers
2. Integrate, unify and transform data
3. Reduce the dimension of the data set ? (eliminating not useful features)
4. Feature Extraction? (make 1 feature out of 2 ) Do underlying, latent variables exists?
5. Data cleaning: split fields, remove spaces, fix format. (+ there are duplicates as some tweets contain both #Trump and #Biden)

Did the data quality change?

Do the instances form natural groups and if so, how can these groups be described?

(check Hopkin’s statistics?)

1. **Modeling.**

Which tools were used: Python

Special libraries in Python: geopandas, sklearn, pandas

Which models were built: clustering (unsupervised learning)

Questions: What kind of model should we use and why? What is the best technique to build the model? Why do we use Python?

1. **Evaluation.**

It is needed to gain confidence that results are valid and reliable and ensure that the model satisfies the original project goals.

* Illustration of work on a poster
* Present the poster (video) including Q&A

Questions: Did we achieve project goals? Is this model good enough in terms of project requirements? Do the results obtained match the real system?

*“One 2017 study explored the link between political liking behaviour and actual voting intention, and found that liking politicians’ public Facebook posts can be used as an accurate measure for predicting voter intention.”*

1. **Deployment.**

Models are put into real use in order to realize some return on investment.

Questions: How is the model best deployed?

1. Sloan L. et al. Who tweets? Deriving the demographic characteristics of age, occupation and social class from Twitter user meta-data //PloS one. – 2015. – Т. 10. – №. 3. – С. e0115545.

   +

   BBC trending [↑](#footnote-ref-0)