COVID 19 Tweets

EDA | Topic Modelling

Venugopal Bhatia

Contents

1. The Data

1 a. Description

1 b. Some Basic Statistics and Insights

1 c. FAIR Principles

1 d. Data Cleaning and Preprocessing

1. Exploratory Data Analysis

2 a. Tweets over time

2 b. Word Cloud

2 c. Term Frequency

2 d. Mention Frequency

2 e. Tweet Length Distribution

1. Sentiment Analysis
2. Topic Modelling
3. Web Application
4. What picture does the data paint?
5. The Data

1a. Description

The dataset that was used for this project was derived from Kaggle. It happens to be one of the more popular datasets primarily used for NLP using covid related tweets.

It has about 41157 tweets along with their location, date, and a manual annotation of tweet sentiment that varies between Extremely Negative to Extremely Positive.

The UserName and ScreenName were Anonymized by the author for compliance reasons.

A summary of column names and what the data looks like is as follows:

* 'UserName’: Anonymized
* 'ScreenName’: Anonymized
* 'Location’: London, London, UK, California, Sacramento, CA, worldwide
* 'TweetAt’: Dates
* ‘OriginalTweet’: 'As news of the region\x92s first confirmed COVID-19 case came out of Sullivan County last week, people flocked to area stores to purchase cleaning supplies, hand sanitizer, food, toilet paper and other goods, @Tim\_Dodson reports https://t.co/cfXch7a2lU'
* 'Sentiment’: Extremely Negative to Extremely Positive

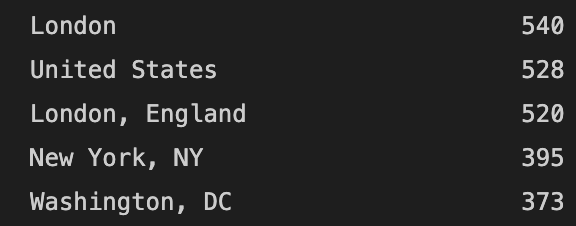
1b. Some Basic Statistics and Insights

Here are some basic statistics about, and insights into what may have happened:

* The location column was by far the dirtiest. A lot of the locations were missing and/or some random string which indicates issues with possibly the way these tweets were acquired.

I do have a theory as to what may have happened here but no evidence to back it up though. My guess would be that perhaps there was web scraping done and that the location may have occurred under different html tags and so the code didn’t pick up the location or collected incorrect values. I’ve experienced something like this first hand while web scraping and this looked very similar. This is conjecture but I’d have to say this is MCAR since I couldn’t find any such pattern and only looking into how data acquisition was done is the only surefire way of knowing what happened.

* 21% of the location column is null. Some location values are not locations at all. 1% of the tweets were from London and there are 12220 unique locations. Here are the top location values by frequency:



* The data was for the period of Jan 2020 to Dec 2020 and it was for the 4th of every month, except for March and April where there was a lot more data. Again, my theory is that maybe since lockdowns happened globally in the month of March the author may have decided to acquire more data during that period, the first date after March 4 is then March 16th and India went into Lockdown on the 24th so there may be some association there but again this is conjecture.

1c. FAIR principles

Our data follows the FAIR sharing guidelines:

1. Findability: The dataset is quite popular on Kaggle and pretty easy to search for and retrieve.
2. Accessibility: Kaggle allows users to retrieve the data either by downloading it from the website or through their API.
3. Interoperability: Data and metadata are in csv format. The comma separated values format is quite common and easy for the computer to read and parse.
4. Reusability: The dataset has a license, Data files © Original Authors and is free for all to reuse. The Kaggle webpage makes it clear that the data is free to use for all and there are no other compliance norms specified on the webpage.

1d. Data Cleaning and Preprocessing

**Location**: The location column was the dirtiest, in order to make the column usable I removed any values with less than one character, cleaned special characters and added NaN for missing values.

Initial intent was to use a mapping to standardize the data since one may have formats such as just the state, or city and state, or state and country, etc. But given the messy nature of the data in the column that task proved increasingly difficult.

I resorted to manually correcting as much of the column as I could and implemented a searchable multi-select in the dashboard. The multi-select displays locations with a frequency of at least 20 though this can be adjusted.

**Dates**: The dates were pretty clean. There were just two dates missing which I filled with NaT.

**Tweets**: For the tweets I used regex to extract mentions, links and hashtags. In the cleaning part the regex also removes any special characters or characters that are not from the English language.

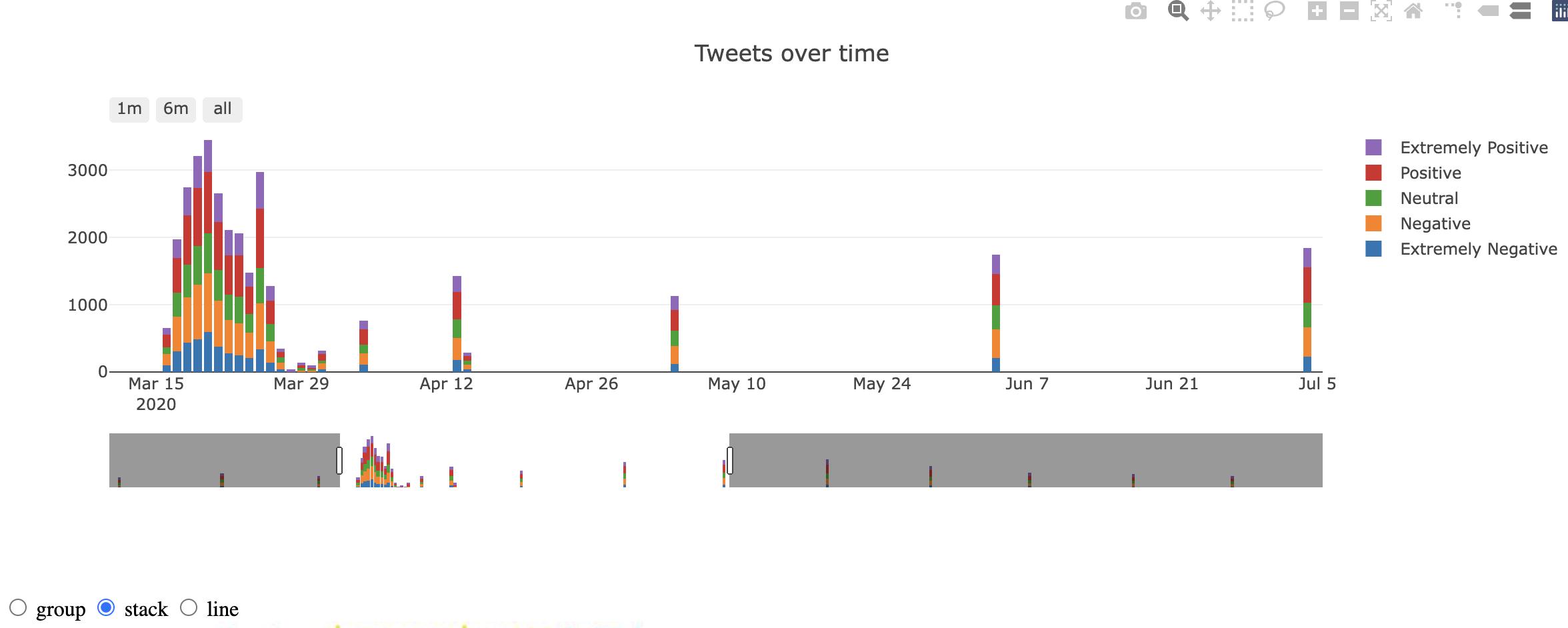
In order to do the Sentiment Analysis, the tweet column was subject to further treatment.

1. The cleaned tweets were tokenized, i.e., we created word tokens.
2. From these word tokens, stopwords like all, they, the, an, for which occur most frequently and convey little to no information were removed.
3. The remaining tokens underwent a Part of Speech (POS) Tagging to understand whether they were adjectives or verbs or adverbs, etc.
4. The Token and POS tag pairs were used for lemmatization to bring the words down to their base form or lemmas.
5. Exploratory Data Analysis

For EDA of the tweets, I did the following visualizations:

* For a subset of regions one can look at tweets over time,
* The word cloud for each of the sentiments,
* The n most frequent words for that sentiment where one can select n
* Top mentions by threshold count in that subset
* Histograms for distribution of probability densities of tweet lengths for each sentiment in that subset of the data.

2a. Tweets over time



For tweets over time one can visualize the data as a group bar, stacked bar or a line graph. This and all visualizations were done using plotlyJs. The slider at the bottom is a time range slider which enables one to look at the tweets and their sentiment for a particular time range. Clicking on just one of the sentiments shows the data for that sentiment only and hovering over the visualization shows counts of tweets.

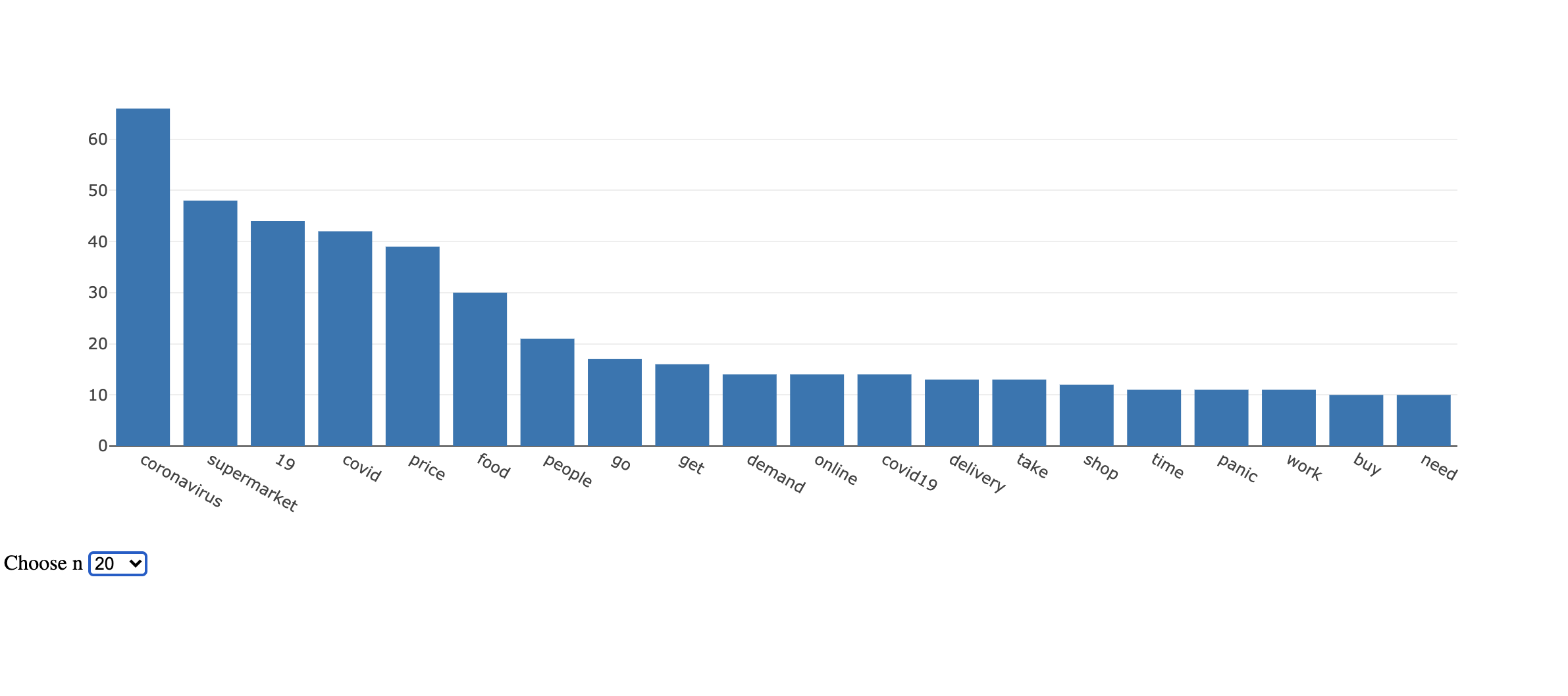
2b. Word Cloud



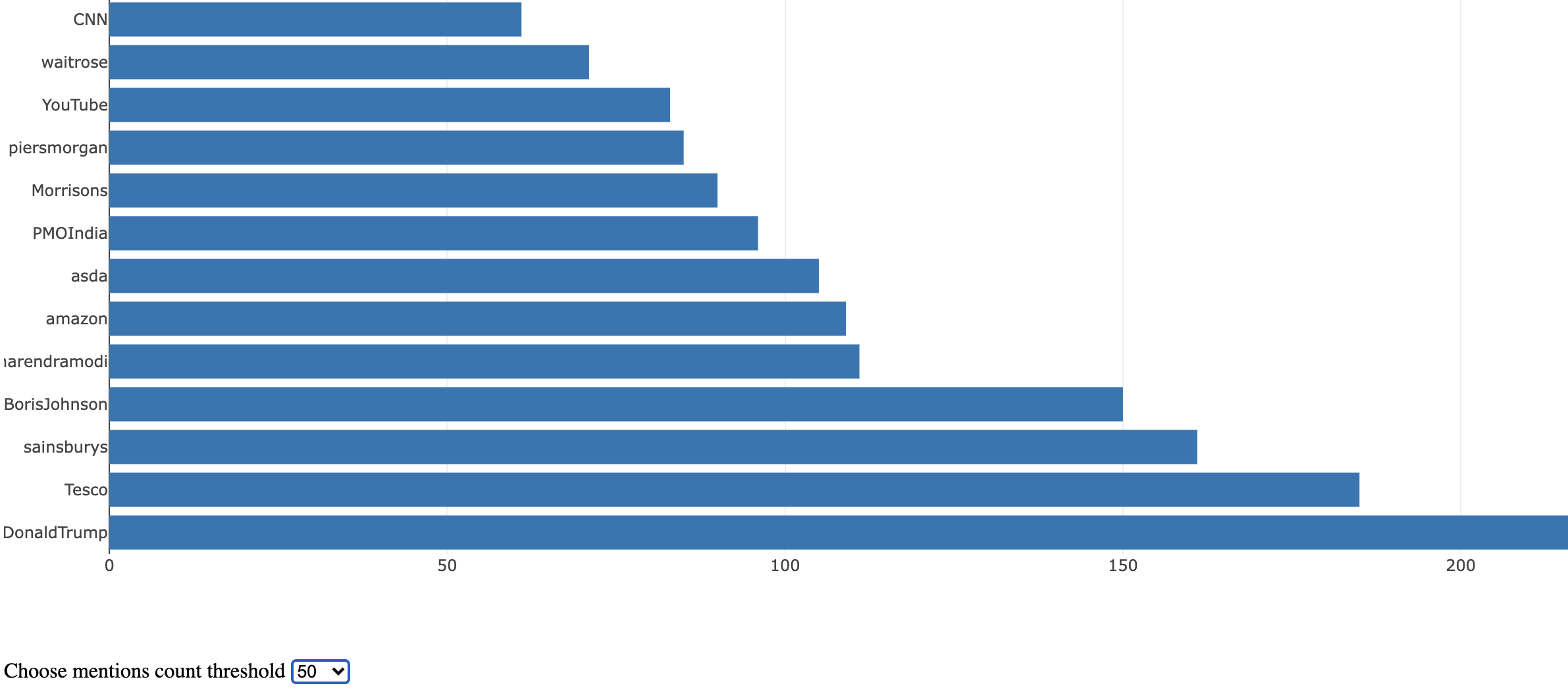
For the subset of regions selected at the top, one can then look at the word cloud for each of the sentiments. The word cloud images are generated using the wordcloud library and an ajax request basically changes the link to the image on the frontend, deleting the previous wordcloud from the images folder. The wordcloud has been set to a maximum of 10000 words and 500 px (height) by 1000 px (width) dimensions.

2c. Term Frequency

One can look at the top 10, 20, 50 or 100 terms by frequency and sentiment. The sentiment here is determined by the radio buttons for the word cloud, thus as the threshold is changed an ajax request updates the plotly graph and if the sentiment is changed or the subset of regions are changes, the graphs are reset; the new top 10 terms for that sentiment, or for Neutral in case regions were changed, are fetched.

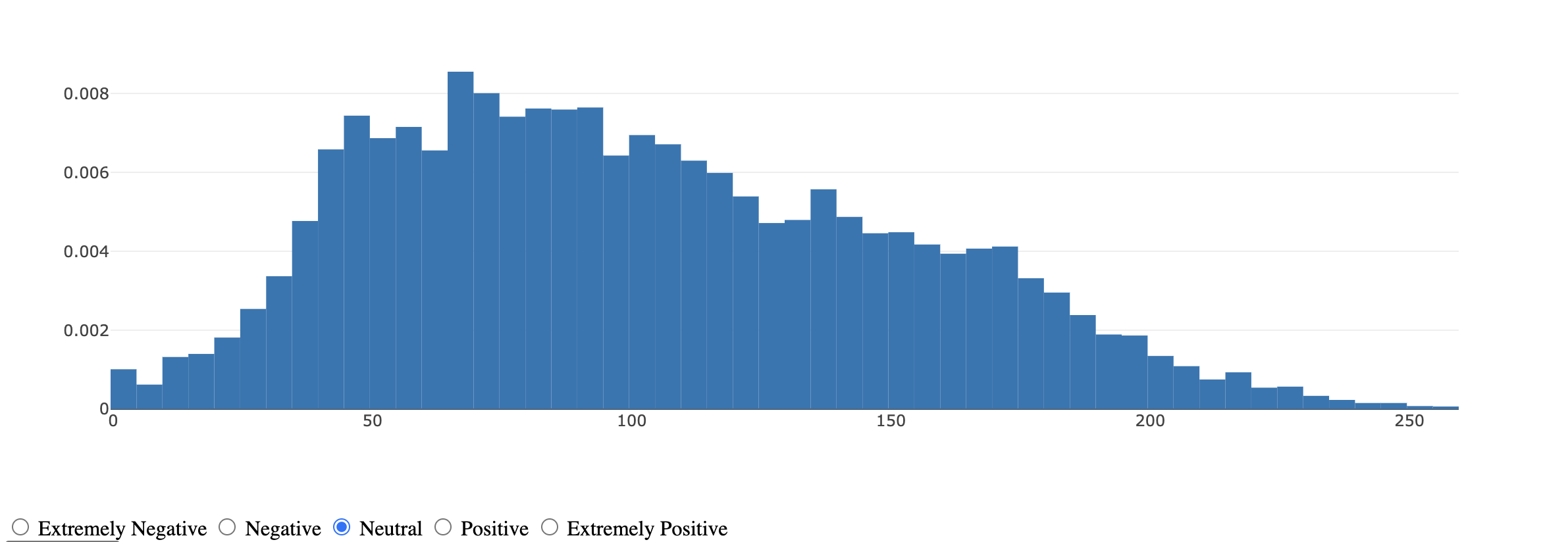


2d. Mention Frequency

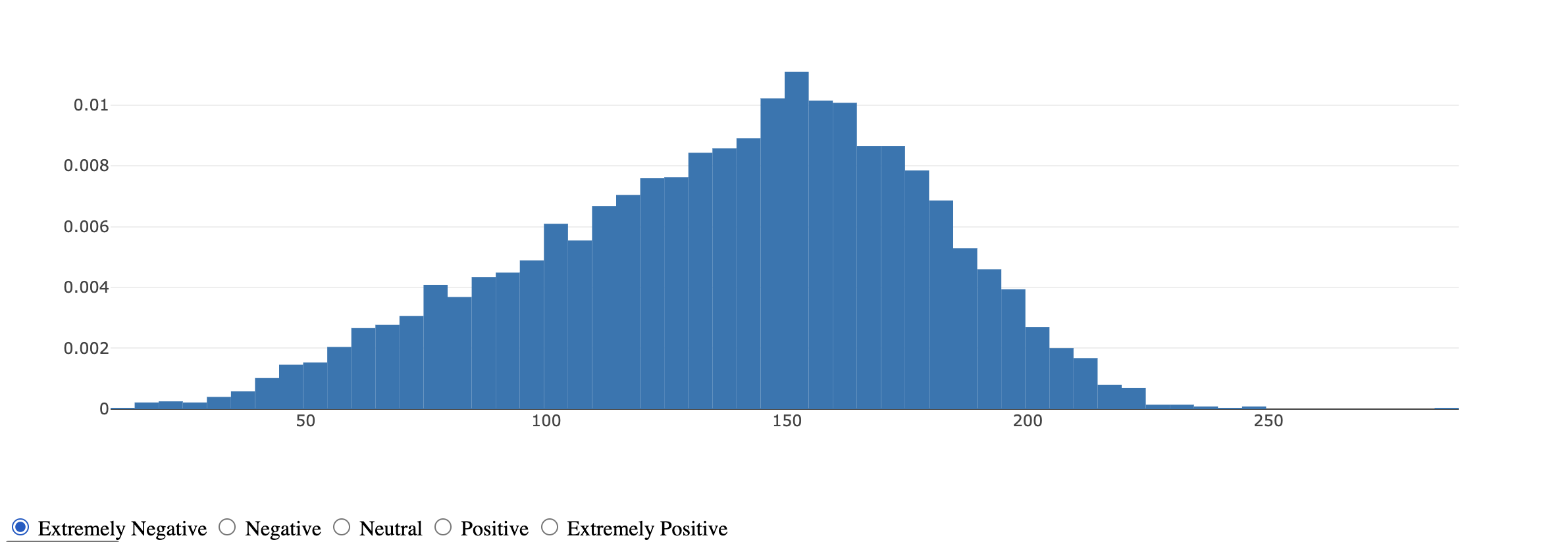


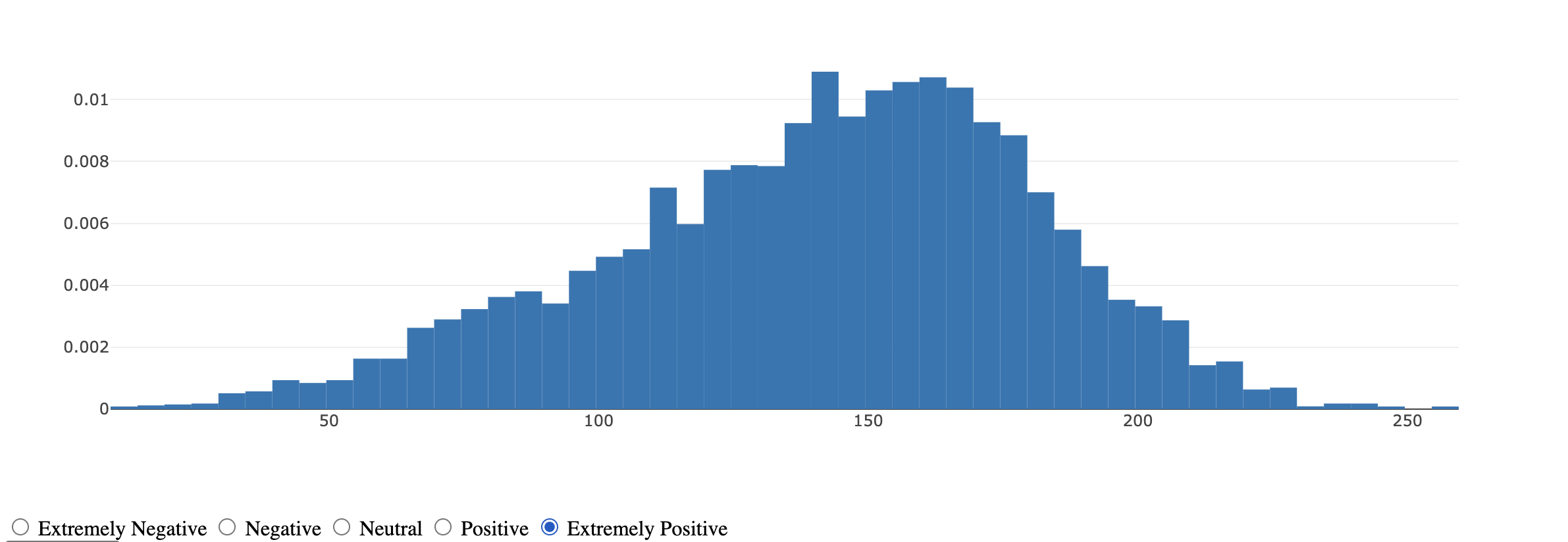
Similar to top terms, one can also look at top mentions, this however is simply the mentions for the subset of regions and not by sentiment. As the regions are changed, the top mentions change as well. Here there is a selection for threshold i.e., the minimum frequency for the mentions, which one could set at 10, 20, 50 and 100. The default is 100 so when regions are reset only mentions with a count of at least 100 show up.

2e. Tweet Length Distribution

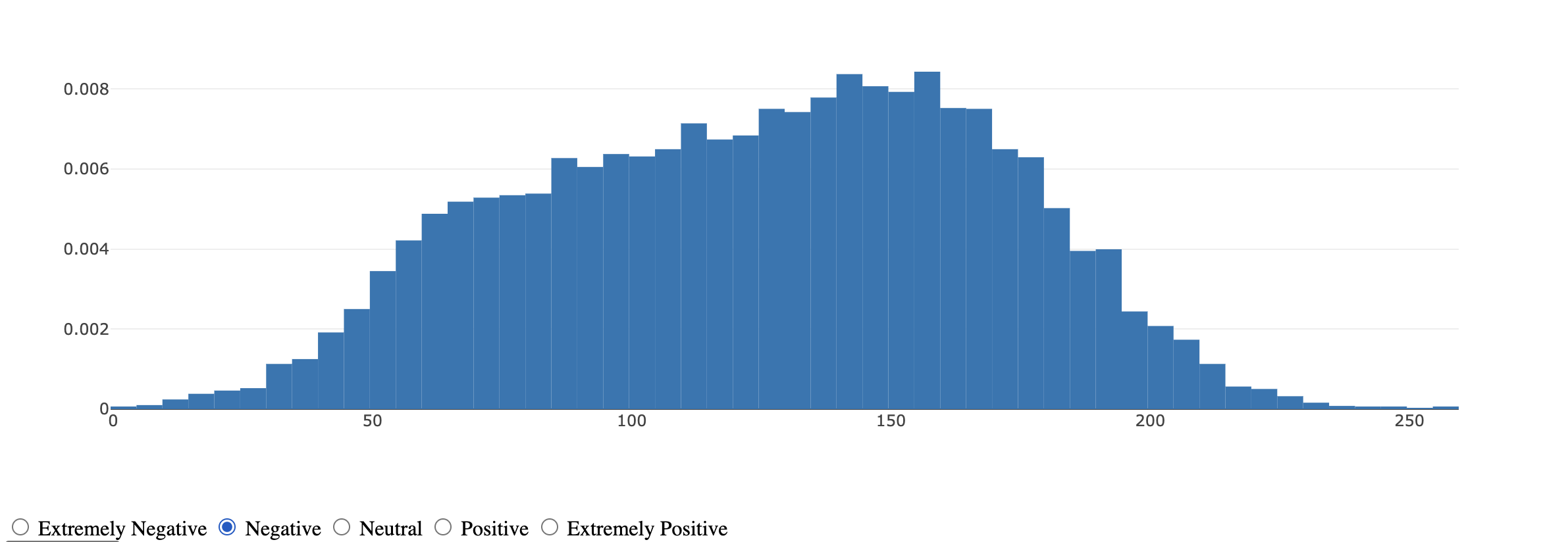


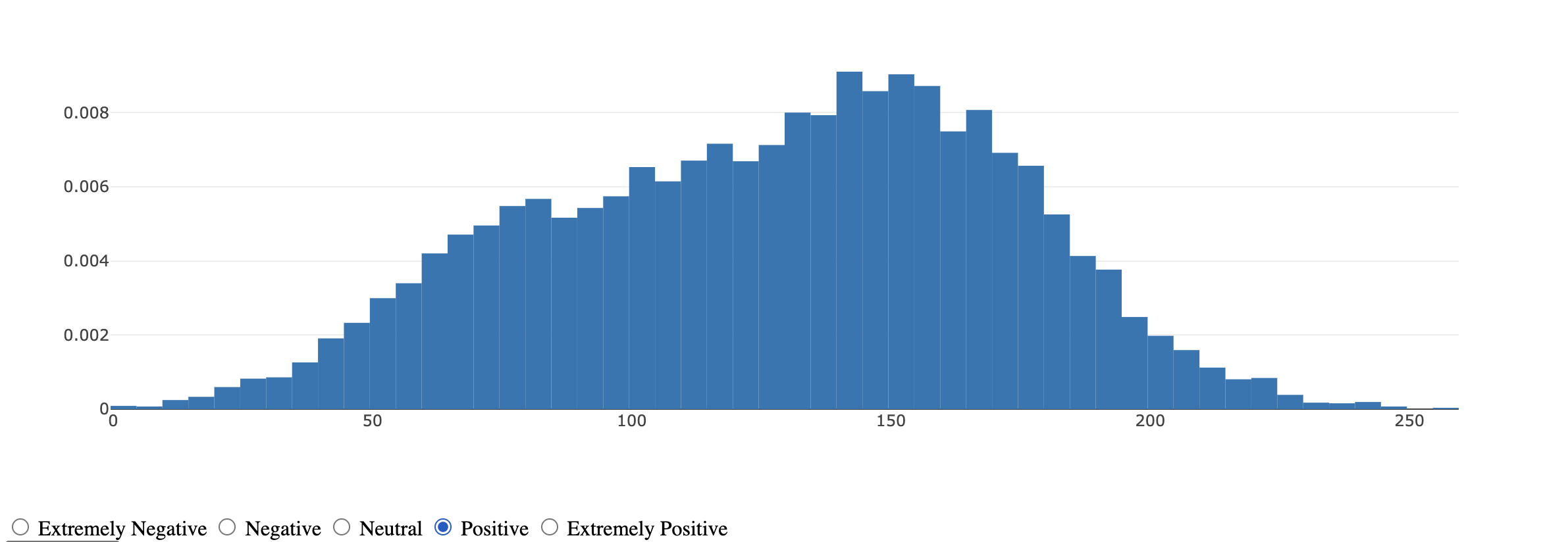
*Neutral Tweet Length Distribution*





*Extremely Negative versus Extremely Positive Tweet Length Distribution*





*Positive versus Negative Tweet Length Distribution*

There is also the ability to look at the distribution of the probability density of the length of the tweets by sentiment and region. This gives a vague idea of whether the length or character count of tweets vary by sentiment.

What I found interesting here was that the neutral tweets have a right tailed distribution and that the extremely positive and extremely negative sentiment tweets have a similar distribution and the positive and negative sentiment tweet length histograms look similar.

Perhaps people have a natural tendency to write 160 characters but the neutral tweets look distinctly different and this would be indicative that while positive and negative tweets could span a wider range of character lengths, the extreme sentiment distributions distinctly tend to be lengthy.

1. Sentiment Analysis

The initial intent here was to try and do Sentiment analysis in order to have an ensemble of classifiers that could then predict the sentiment of covid related tweets.

To this end I used the CountVectorizer to get the lemmatized word tokens as features this is then followed by conversion of the lemmatized tweets to vectors with feature counts in each tweet.

For hyperparameter tuning the conventional methods employed usually use GridSearchCV or a random walk over the hyperparameter space. I intended to use Markov Chain Monte Carlo (MCMC) to solve this problem, so basically having a prior and a likelihood function and then calculating posteriors which would help determine the hyperparameters.

I used a library called Hyperopt which does this Bayesian Inference and uses an algorithm called Tree of Parzen Estimators instead of MCMC. I assigned uniform priors for each of the hyperparameters for each of the algorithms.

However, this analysis didn’t give very good results, probably since discerning the extremely positive from positive and vice versa is difficult and so for the purposes of our assignment I abandoned this analysis

1. Topic Modelling

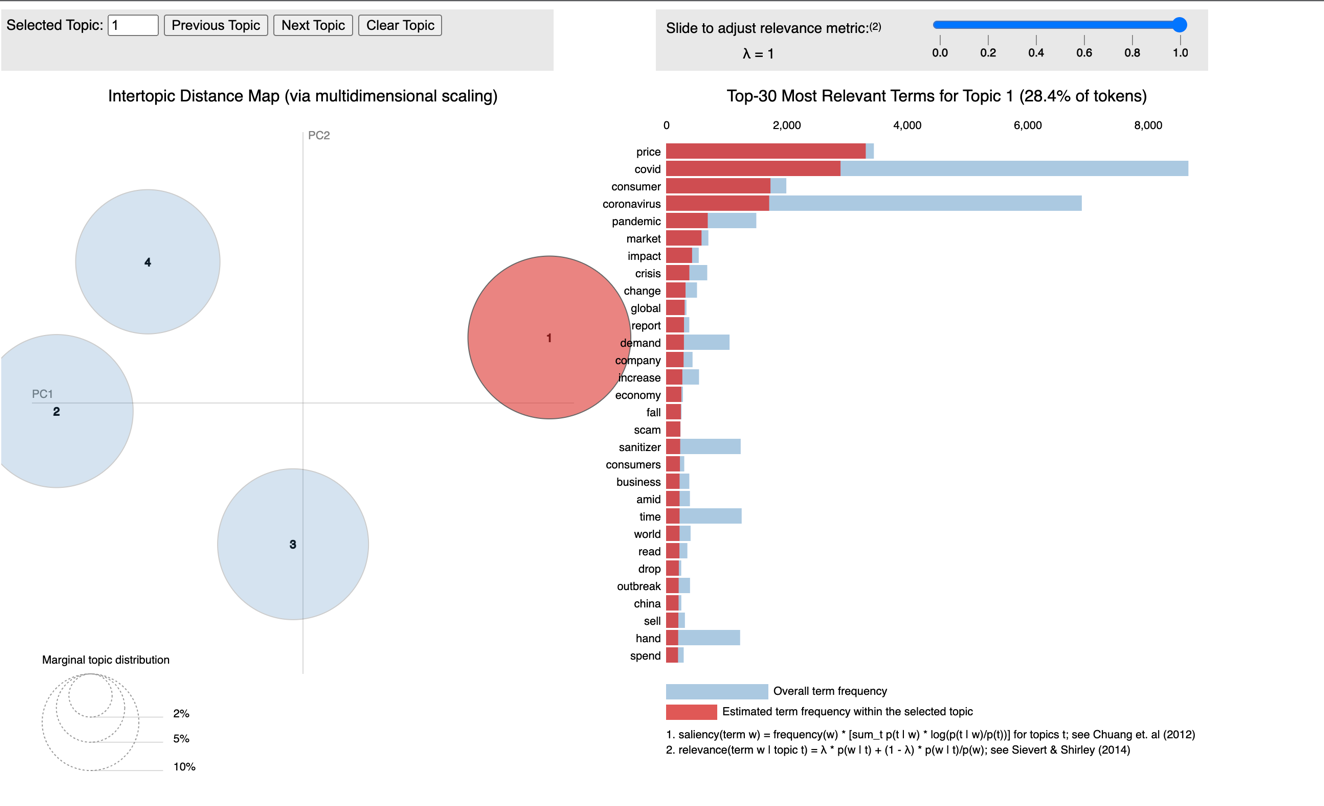
For Topic Modelling I employed the Latent Dirichlet Allocation model which basically uses Bayesian statistics to determine abstract topics and corresponding words with weights assigned to each topic. The model was implemented using Gensim on the lemmatized tweet data.

I tinkered with various number of features starting from 10, but at 10 a lot of the features had similar words and weights, I got the following result with 4 topics:



Looking at the words in Topic 0 one could say people were panic buying groceries, Topic 1 seems to hint at demand for workers and help at stores was affected.

Hand sanitizers, masks and online shopping seem to be the words for Topic 2 and Topic 3 talks about consumers, global markets, prices and impact of the crisis.



*The Topic Model visualization*

1. Web Application

The Web Application was setup with a Model View Controller (MVC) type architecture in Flask. The folder controllers has the file for the various methods of querying the dataframe while the model has various .py files required for preprocessing cleaning, hyperopt, and so on.

There are two interactive notebooks as well (analysis\_cleaning.ipynb and insights.ipynb) that have in them the various steps that were used in the EDA as well as the results. The data folder manages all the csv files, which means any files generated during or after any sort of analysis are also stored in that particular folder.

The static folder has the images, javascript code and any css while the templates have the file for the homepage with the EDA and a separate file for the LDA based topic modelling visualization.

1. What picture does the data paint?

To summarize what I have inferred from this dataset with the analysis so far:

* It seems to me like maybe more data collection around March may have been done on purpose since that’s when lockdowns happened.
* Top mentions included media personalities and online shopping platforms such as DonaldTrump, BorisJohnson, NarendraModi, PiersMorgan and also amazon, Sainsbury and Tesco.
* The pdf histograms of the tweet lengths also were kind of interesting in the sense that the distributions for extremely positive and extremely negative looked similar, so did the distributions for positive and negative sentiment, and all looked different from the neutral tweet length distribution.
* The Topic model reflects what was on people’s minds: Loss of jobs, Panic Buying, Supply and demand disruptions and effect on global markets, online shopping of masks, hand sanitizers, supplies and groceries.