2020 US Election Twitter User Analysis

Gengxing Wang, Yujia Zhang

1 Introduction

Since its establishment back in 2006, Twitter has been widely considered as one of the most popular social media platforms. On average, there are approximately 353 million monthly active users (MAU), generating over 15 billion tweets every month. Such a huge volume of data intrinsically comes with great immediacy due to the characteristics of Twitter, making it feasible for data scientists to conduct natural language processing tasks and acquire informative insights for events with popularity.

During the US election in 2020, Twitter has drawn great attention due to its extensive usage by both parties. Massive tweets that are relevant to the election were posted in particular a few weeks before the election. This motivated us to think about whether it is possible to conduct extensive analysis on the corresponding data and discover the difference of supporters for two parties respectively. In this project, we aim to build a natural language processing pipeline to analyze the tweets with certain hashtags that are relevant to the 2020 US election. Our objective is to provide a comprehensive analysis of the composition of users who posted those tweets concerning multiple factors such as geographical location, tweetness, and utility used for tweeting. More than that, we also aim to compare the two groups, analyzing the discrepancies, and see if the result matches the common sense and actual election result. As a result, our trained sentiment analysis model matches the actual election result in most of the states, as shown in Fig. 1

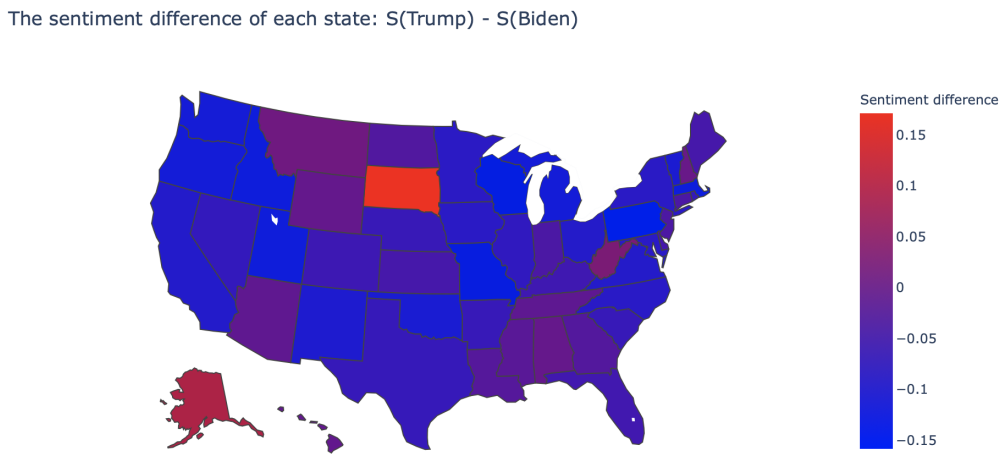

Figure 1: The preference of each state in the US using our sentiment analysis model. Red indicates that the state favors Donald Trump more, while blue indicates the opposite.

The rest of the report is organized as follows: in Sec. 2, we describe the two datasets we employed in our project, including some basic statistics and why we decided to employ them. In Sec. 3, we elaborate the preprocessing procedures taken on both datasets for raw data cleaning and feature engineering, as well as the machine learning model we experimented with and eventually employed for sentiment analysis. We report our results in Sec. 4 from multiple aspects, and summarize our project in Sec. 5.

2 Data

In this project, we employed two datasets for different purposes. [4] is the dataset we conduct all of our major analysis on. Originally, it comes with over 1.7 million tweets collected a couple of weeks before the election day (Nov. 3, 2020). The dataset consists of two files for two candidates respectively, where the first one is for tweets with tag $\#Trump$ or $\#DonaldTrump$ and the second one is for tweets with tag $\#Biden$ or $\#JoeBiden$. There are over 20 attributes for each data entry, including the information at geographical level (state, city, coordinate, etc.), user-level (user name, number of followers, etc.), and tweet-level (how many retweets and likes, etc.). We observed a high missing rate for certain columns. After only preserving tweets with at least one geographical information and were in English, we have reduced the number of available data entries to 840499 (363894 for Biden and 476605 for Trump). We conduct most of our experiments on the filtered dataset.

The reason that why we decided to utilize an external dataset instead of collecting it by ourselves is that there exists a monthly pull restriction for free Twitter API users. Each user was also restricted to pull only 500,000 tweets each month, which we thought could be insufficient. After a thorough investigation and requesting approval from Prof. Choudhary, we decided to use an external one for the most reliable result.

Since the US election dataset [4] does not contain any label for sentiment analysis, we employ an extra dataset with binary sentiment labels, namely sentiment-140 [1] for training our machine learning model. [1] originally comes with approximately 1.6 million tweets, with each labeled as either positive or negative. It is also one of the most popular datasets for sentiment analysis purposes. Note that while the dataset [1] consists of tweets from different fields and various topics, the [4] concentrates more on the US election and thus the vocabulary does not necessarily overlap perfectly, which could result in inaccuracy. Nevertheless, [1] is the best dataset that we found handy, and the actual outcome is also promising eventually.

3 Methods

3.1 Data Cleaning

Since the raw dataset [4] collected still contains an unneglectable amount of noise, we apply the following steps for data cleaning. We applied the same data cleaning procedures for both two datasets. In order, the procedure taken were:

1. Language filtering. Due to the linguistic difference, the sentiment analysis model trained on one language is not necessarily transferable to another language. As all

tweets from sentiment-140 [1] are English, we keep tweets in English only. This is achieved by using langdetect [9].

2. Removing entries with certain attributes missing. As our final objective is to acquire insight and visualize them based on the predicted sentiment, it is essential to ensure all the entries have the corresponding attributes. Specifically, we preserved entries with at least one geographical attribute and also the utility attribute available. This is achieved by using regular expressions.

3. Tweet text cleaning. Since the original tweets consist of not only alphabetical letters but also emojis, links, and all other types of characters, we preserve the part written alphabetical letters only. Such cleaning can reduce the sparsity as well as increase the likelihood of matching to the correct word that appeared in the training dataset. Only tweets with over 10 characters after filtering were preserved for further analysis.

3.2 Feature Engineering

Encoding is also required to convert the text data into representation meaningful for mathematical models to interpret. In our project, we employed TF-IDF [8] as the approach to encode the text data into features. There are two terms calculated in the TF-IDF, respectively the word frequency tf:

$$tf(t,d) = log(1 + freq(t,d)) \tag{1}$$

and inverse document frequency idf:

$$idf(t,D) = log\frac{N}{count(d \in D : t \in d)} \tag{2}$$

where t denotes the corresponding word, d denotes the current document and D denotes the document set. Combining the two terms, the final tf-idf score is calculated as:

$$tf - idf(t,d,D) = tf(t,d) \times idf(t,D) \tag{3}$$

By doing such a calculation, we can convert the text data into numbers in arrays. Nevertheless, the sparsity of the resulted matrix is determined by the number of the total appeared words in the dataset, and consequently, our processed feature is over 2,000 dimension, which theoretically limits us using models that are not capable of handling high-dimension inputs conveniently (such as SVM with no kernels).

3.3 Sentiment Analysis

We employed two types of sentiment analysis approaches, respectively vocabulary-based and machine learning-based. We firstly have tested the feasibility of two vocabulary based methods, respectively TextBlob [6] and [5] from NLTK [2]. Both two methods depend on a pre-collected dictionary, matching each word from the input text, obtaining the polarity, and summarizing them as the final output score. While the method is fast, it fails to generalize well on unseen and informal datasets including a massive amount of slang and abbreviations.

The second category is machine learning (ML) based. One significant advantage of machine learning methods, comparing to traditional methods, is their adaptability to arbitrary datasets. While traditional methods rely heavily on manually crafted features, ML methods can craft the most optimized features by themselves during the process of training. We experimented with some of the most popular ML methods such as logistic regression, support vector machine, and Naive Bayes from scikit-learn [7], as well as one of the most advanced ensemble-based method XGBoost [3]. As shown in Tab. 1, the logistic regression model we trained achieved the highest performance on [1], and we decided to use it for all subsequential analysis.

For the logistic regression we employed, we applied L1 regularization in specific to handle the sparsity within the process of the dataset. Binary cross-entropy was used as the cost function, which results in the following objective function:

$$\min_{w,c} ||w||_1 + C \sum_{i=1}^{n} log(exp(-y_i(X_i^T w + c)) + 1) \tag{4}$$

Dataset/Model	Logistic Regression	Linear SVM	Naive Bayes	XGBoost	Vader	TextBlob
Sentiment-140	77.9%	76.0%	75.3%	72.1%	64.7%	62.5%

Table 1: Performance of each method on Sentiment-140 for sentiment analysis prediction.

4 Experiments and Analysis

Following the aforementioned approaches, in this section, we demonstrate the insights obtained via extensive analysis. In Sec. 4.1, we show some analysis results conducted on the raw dataset. In Sec. 4.2, we show results using the predicted sentiments as an extra attribute and attempt to discover certain correlations among those available attributes.

4.1 Statistical Analysis on the Original Dataset

We analyze two datasets respectively, discovering whether the raw datasets are already differing from each other. We first summarize the percentage of the language the tweets are written in. As shown in Fig. 2, 66.8% tweets tagged with *Biden* or *JoeBiden* are in English, which is 3.5% lower than the tweets from Trump. At the same time, there are 1.2% more Spanish tweets for Biden than Trump. We believe this reflects the percentage of supporters for each candidate in some way and matches common sense.

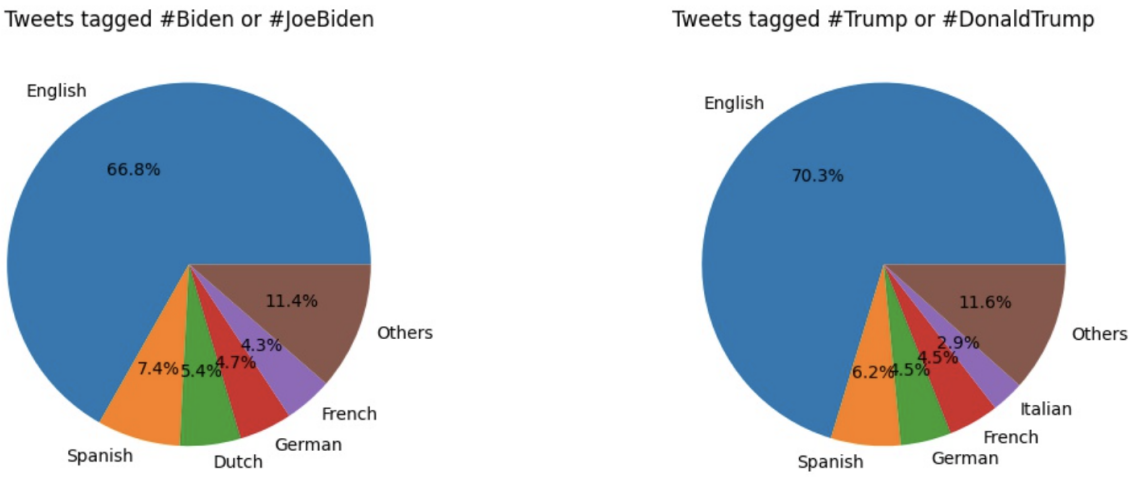

Figure 2: The percentage of languages used for two datasets.

Secondly, we construct two-word clouds for each dataset. As shown in Fig. 3, while there are a few words that are being extremely popular in both datasets such as "vote", "election", "now" and "will", observing, there are still many interesting differences. For example, while "bidenharris2020" appeared on the left, we failed to see "trumppence2020" on the right. The majority of the vocabulary overlaps with each other, yet we still can see the difference in terms of the advertisement strategy based on the subtle differences.

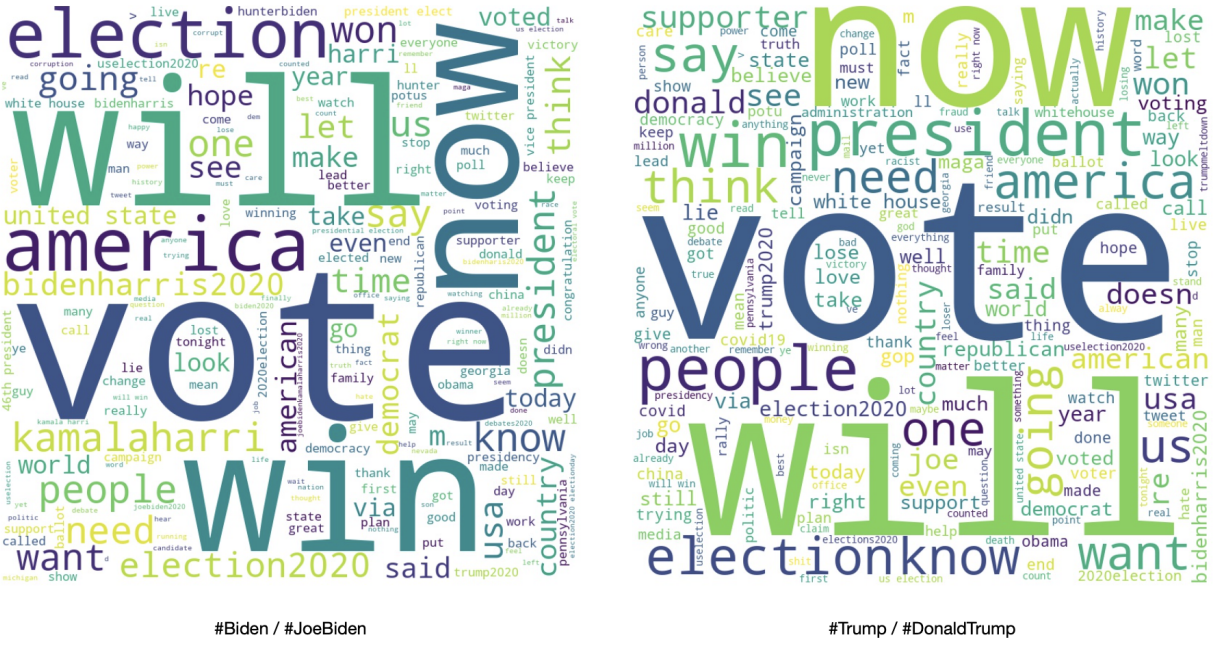

Figure 3: The word clouds for two datasets.

We also summarize the percentage of utilities used for tweeting for each party. As shown in Fig. 4, the primary source for tweets relevant to Biden is iPhone, occupying up to 33.4%.

On the other side, the primary source for tweets related to Trump is using Twitter's web application, also occupying up to 33.9%. There are 4.4% more tweets posted using a mobile device (iPhone or Android) for Biden than Trump, which we believe also reflects the difference of the two groups in some way.

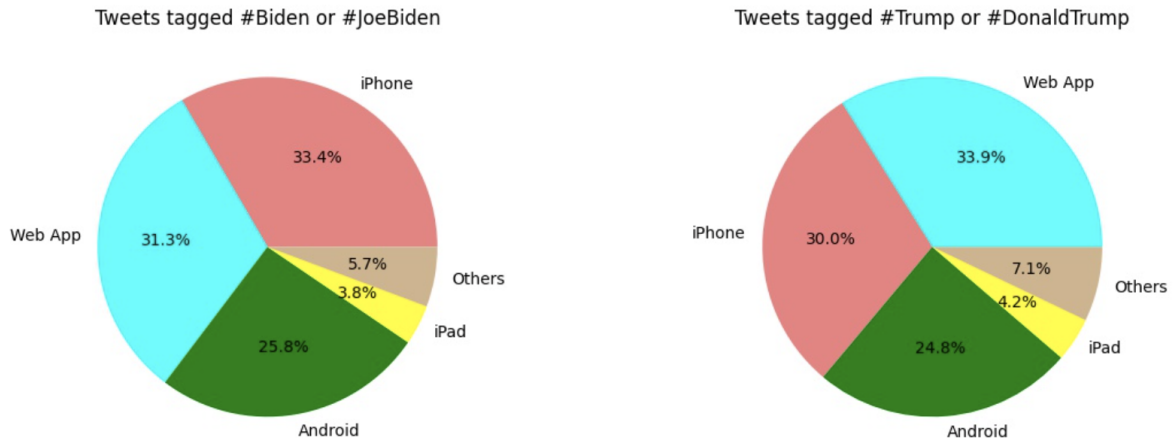

Figure 4: The percentage of utilities use for posting tweets for two datasets.

Lastly, we summarized the geographical location of tweets and visualized it as a heatmap in Fig. 5. The top four states with the most tweets are respectively California, New York, Texas, and Florida, which corresponds to their population precisely. This analysis suggests that the distribution of tweets aligns with the distribution of population well, and there was no outlier state according to our analysis.

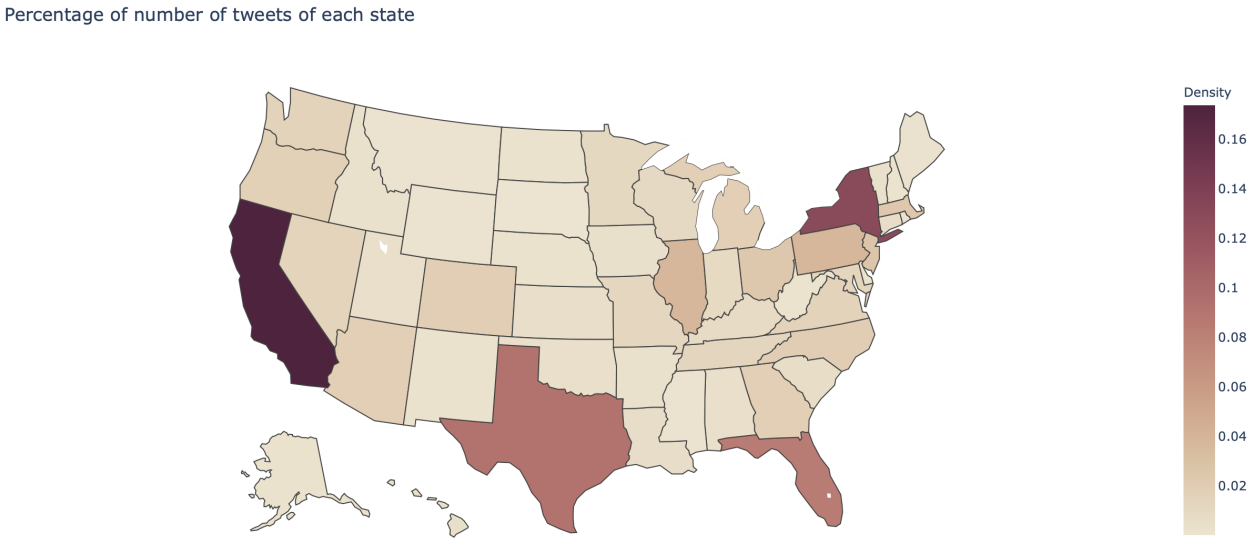

Figure 5: The geographical visualization of the count of where the tweets were posted from.

4.2 Sentiment-based Analysis

We applied the trained sentiment analysis logistic regression model onto [4]. As shown in Fig. 6, the predicted sentiment distributions both follow a Gaussian distribution, and both are skewed towards the positive side slightly. According to our analysis, the average sentiment for Biden (0.24) is slightly more positive than the ones for Trump (0.14), where -1 denotes an absolute negative and 1 denotes an absolute positive.

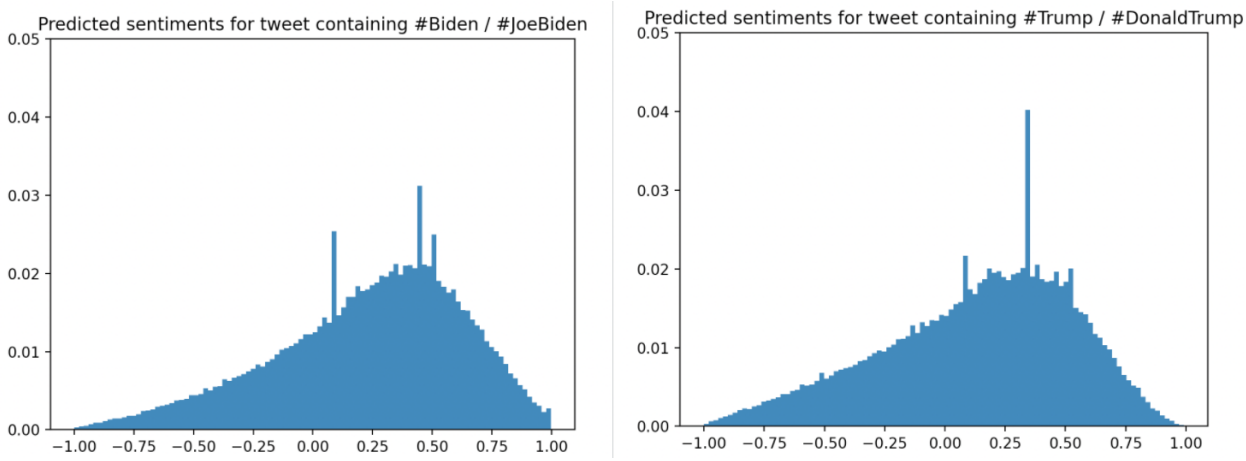

Figure 6: Predicted sentiment distributions for two datasets.

Our second experiment was to discover whether a correlation between the sentiment of the tweet and its popularity (in terms of the number of likes and retweets it received) exists. As shown in Fig. 7, while the correlation is not as strong as we expected, we have still observed a weak positive correlation between the polarity and the popularity. This suggests that people are more likely to like and retweet positive tweets, which also aligns with common sense.

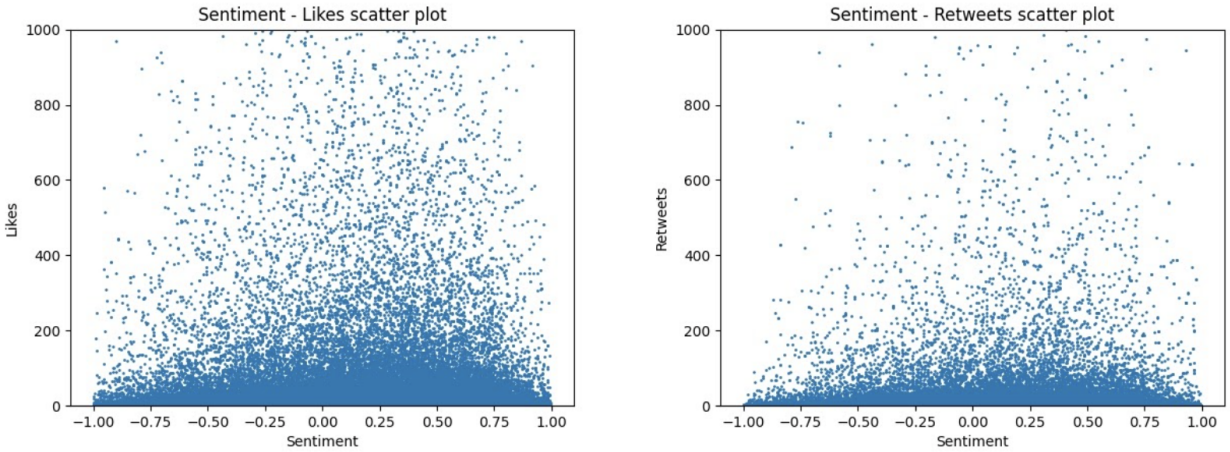

Figure 7: Scatter plots of Sentiment - Likes and Sentiment - Retweets.

We were also very interested in confirming whether the rumor of any of the party was using bots for false advertisement on Twitter. Our strategy is to pick out tweets with over 0.75 polarities (very positive), and plot the statistics of those users in terms of the number of their followers (y-axis) and how many days have those accounts been registered (x-axis). We assume that the bots should have been registered very recently and with fewer followers than others, therefore a large gathering at the bottom-left should indicate its existence. However, we fail to observe such a pattern in both two datasets. We believe that either none of the group was using bots, or both were using advanced bots for advertisement.

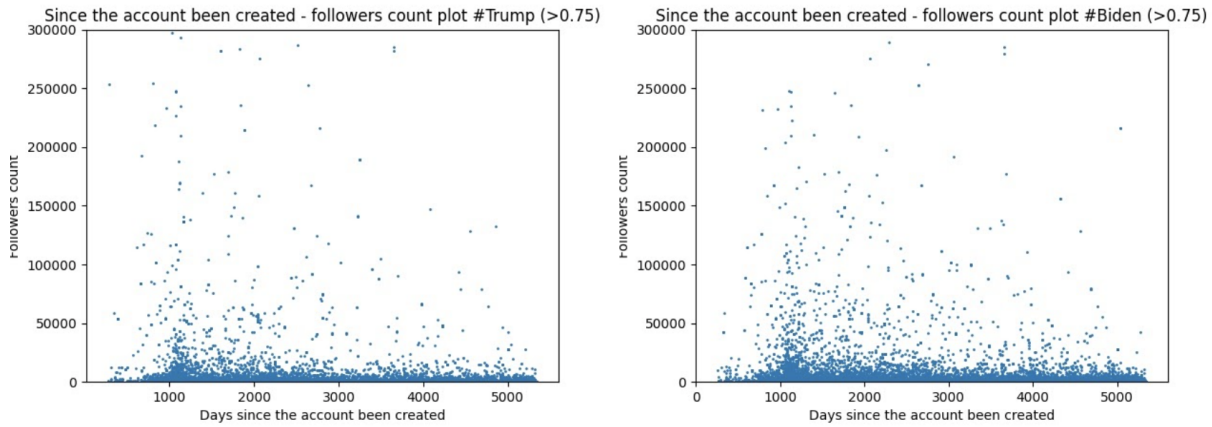

Figure 8: Scatter plots of Days registered - Followers of the accounts in two datasets.

As the topic of the dataset is about US elections, we think it will be very intuitive to compare our results against the actual ones by visualizing the average sentiment at a state level. As shown in Fig. 9, Fig. 10 and Fig. 11, our predicted result matches the actual outcome in general. We observe a dominating advantage for Biden in states such as California, Washington, New York, and Massachusetts, while Trump is dominating in South Dakota, West Virginia, Alabama, and Wyoming. It is noteworthy that the group of Twitter users does not necessarily overlap with the actual American citizens who voted, and we believe supporters for Biden tend to use social media more. Therefore, we believe the visualization is somehow biased.

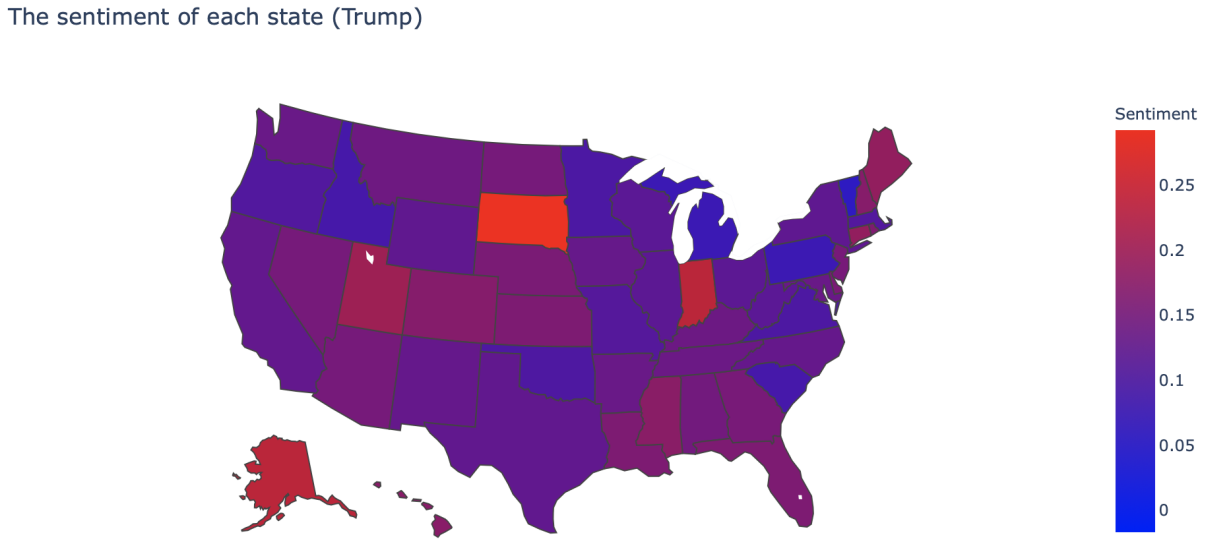

Figure 9: Average sentiment in each state for tweets related to Trump.

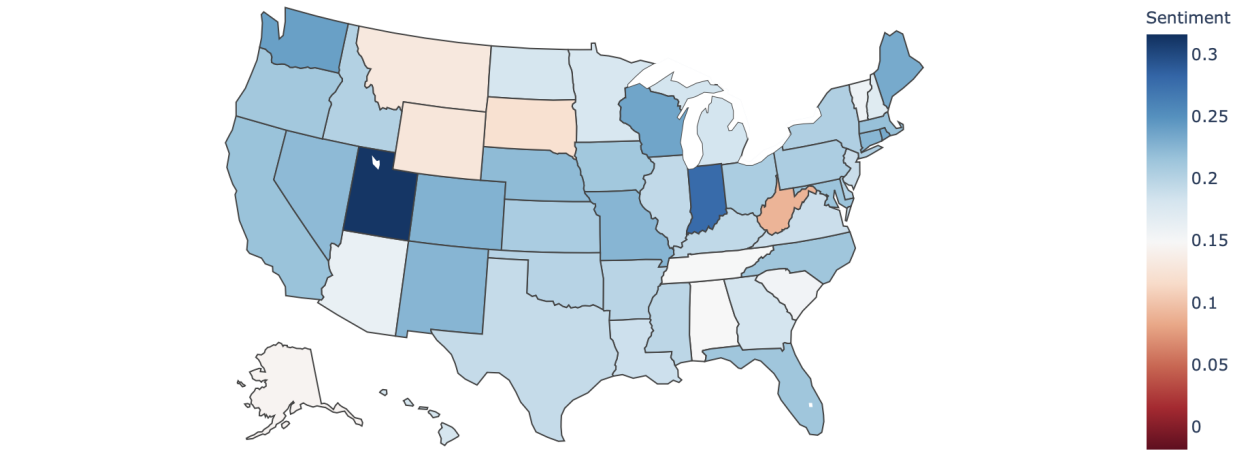

Figure 10: Average sentiment in each state for tweets related to Biden.

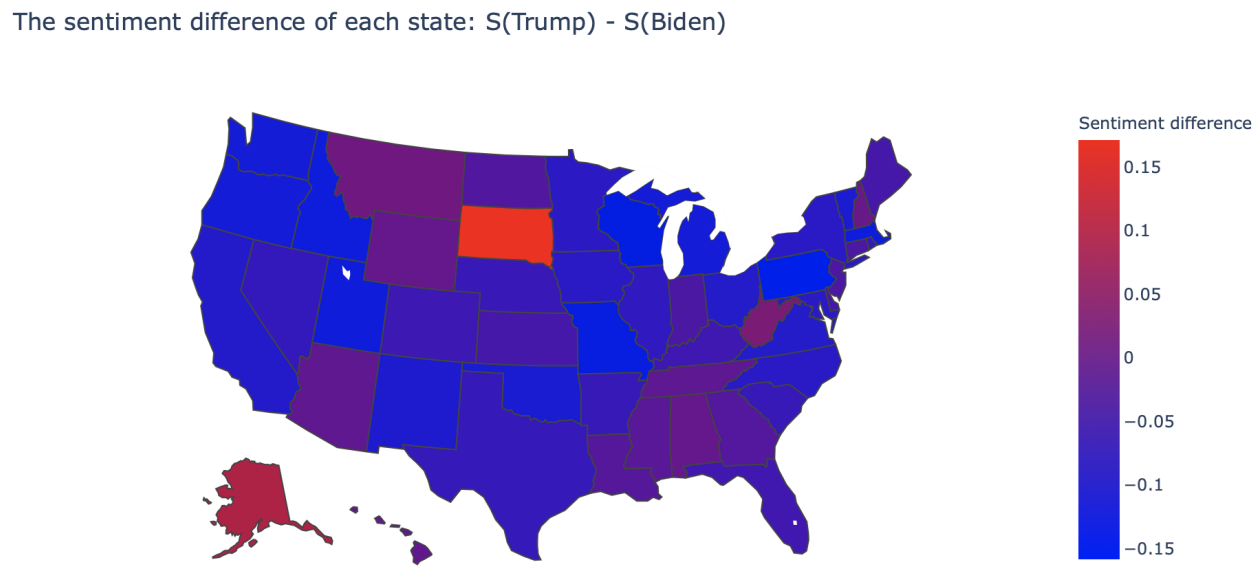

Figure 11: Average sentiment difference in each state.

Following the same method, we extend the scope of the visualization to the world level. As shown in Fig. 12, Fig. 13 and Fig. 14, while most countries are relatively neutral towards two candidates, we observe a strong preference in Russia towards Trump and a similar one for Biden in China. Most of the other countries such as India, Canada, Australia, and the United Kingdom are more neutral comparing to the aforementioned two. From all those analyses, we believe it is possible to acquire and extract further information and for more specific objectives.

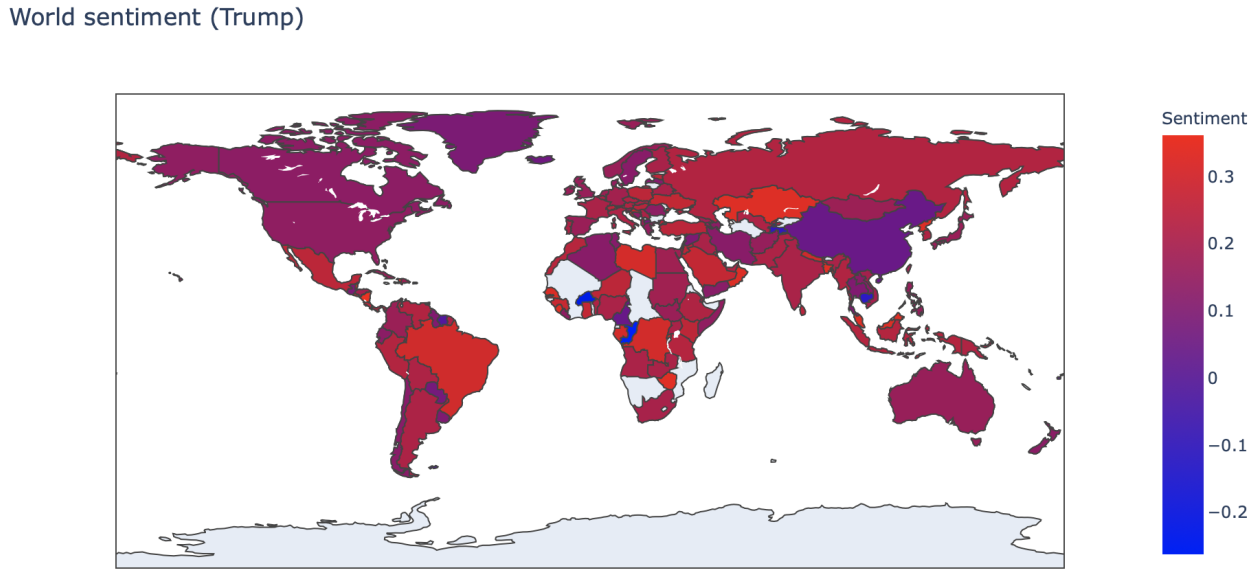

Figure 12: Average sentiment in each country for tweets related to Trump.

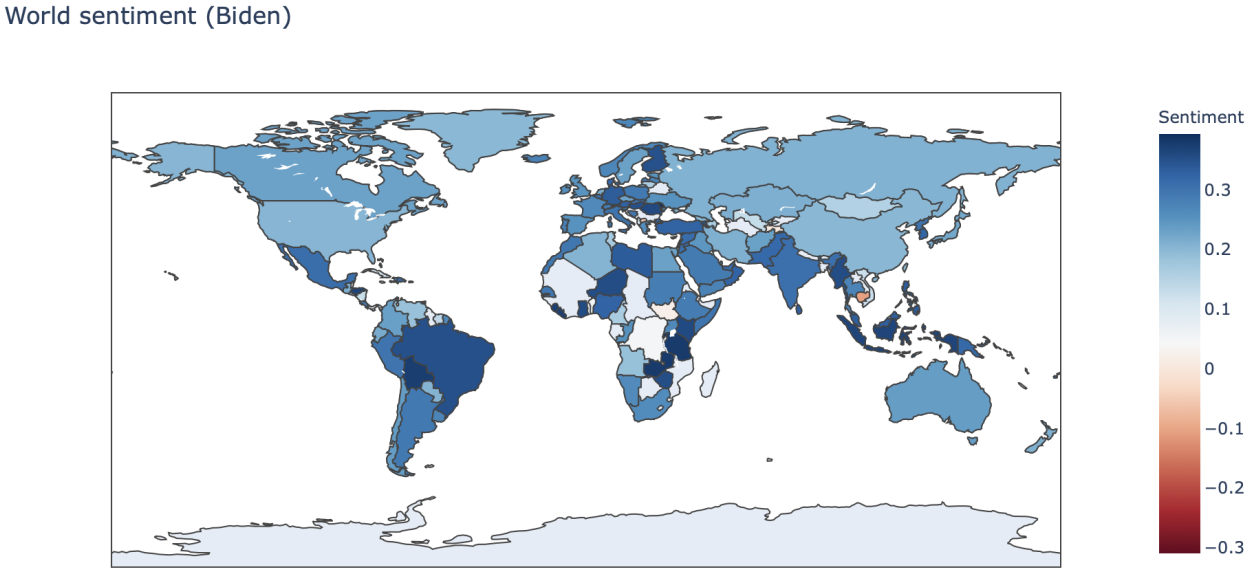

Figure 13: Average sentiment in each country for tweets related to Biden.

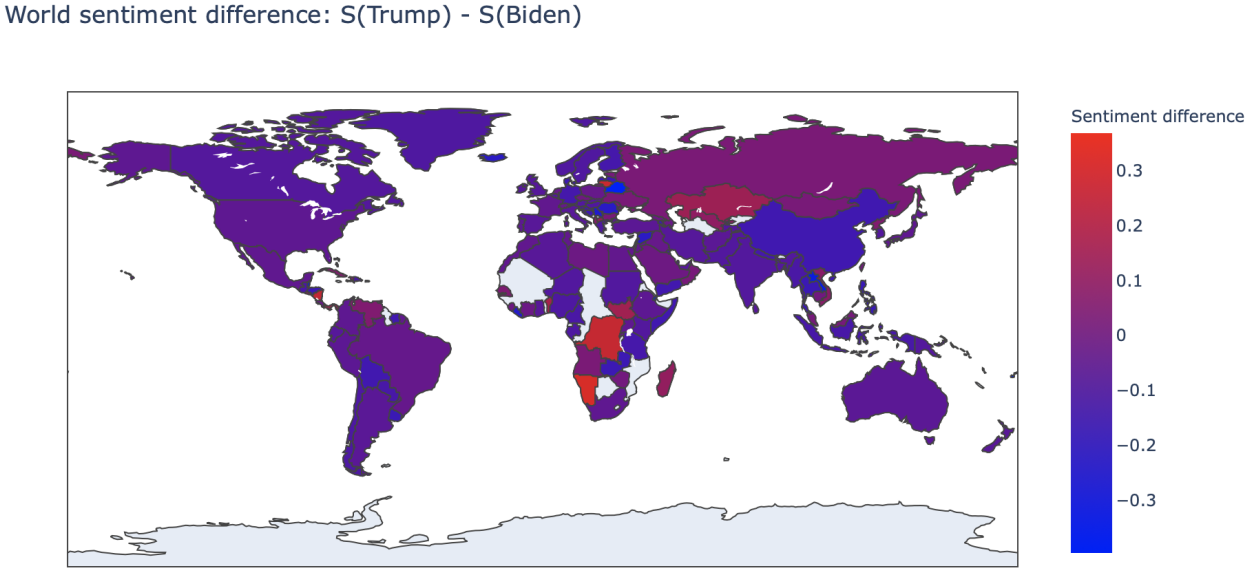

Figure 14: Average sentiment difference in each country.

5 Summary

In this project, we built a machine learning-based sentiment analysis model and performed an extensive analysis on a 2020 US election Twitter dataset. We visualized our findings via different approaches for most intuitive understanding, and have discussed the difference between two groups that appeared in the dataset from multiple aspects. We should note that all the analysis results do not reflect our political standing and we only report what we have observed during the analysis, with no implications to any extent.

We contribute the deviation of the predicted sentiments to four factors: firstly, the distribution and vocabulary of two datasets ([1] and [4]) differ to some extent, and our trained model is thus not guaranteed to generalize well on the unseen dataset. Secondly, due to the extensive usage of slang and abbreviations in the tweets, it is difficult to eliminate their impact on the sentiment analysis part. Thirdly, there also exists a lot of irrelevant posts in the [4] that come with clear polarity, introducing noise into the further analysis. Lastly, the model we chose, while with L1 regularization, is still too naive to handle high-dimension data. For future work, we aim to collect data with higher quality and employ machine learning models that can handle extreme sparsity better. We could also look into more appropriate feature engineering and data cleaning methods to both eliminating the negative effect of noises as much as possible and also obtaining more informative representations.

References

[1] Alec Go, Richa Bhayani, L. H. Twitter sentiment classification using distant supervision. *Standford CS224N Project Report* (2009).

[2] Bird, Steven, E. L., and Klein, E. *Natural Language Processing with Python* (2009).

[3] Chen, T., and Guestrin, C. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2016), KDD '16, ACM, pp. 785–794.

[4] Hui, M. Us election 2020 tweets, 2020. accessed 2016-06-01.

[5] Hutto, C.J., G. E. Vader: A parsimonious rule-based model for sentiment analysis of social media text. *Eighth International Conference on Weblogs and Social Media (ICWSM-14)*.

[6] Loria, S. textblob documentation. *Release 0.15 2* (2018).

[7] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research 12* (2011), 2825–2830.

[8] Sammut C., W. G. Tf-idf. *Encyclopedia of Machine Learning* (2011).

[9] Shuyo, N. shuyo/language-detection, 2009. accessed 2021-06-01.