2-Epoch Temporal Analysis of Election Related Tweets

Michael Telahun

September 17, 2020

Abstract

Twitter data has become a hot topic for data mining tasks in several fields. The unstructured information in Twitter data streams can make the mining processes and analysis difficult to carry out but often results in a rich creation of knowledge. In this work we manipulate, prepare, and ascertain information, with a focus in pre-processing techniques, from two different days, one in August and one in September, in topics related to the 2020 presidential campaign. As such, several data mining pre-processing techniques were used to digest the information deliberately and thoroughly from both days, creating a temporal analysis that could be applied day by day or over any period in the future. The analysis was performed based on three hypothesis that wanted to be addressed with the temporal gap in mind: (1) *interest of one candidate will grow more than for the other in some states,* (2) *more popular people will be tweeting about both candidates,* (3) *sentiment will shift in either positive or negative directions for one of the candidates by 5% or more* The development of these hypothesis and the implementation for the munging of this information from the tweets is explained in detail within this work. In general the results show that this information to some degree is not developed enough to make large assumptions; however, given a larger period or more time periods to compare the same analysis, the assumptions from the results would be more concrete. The results from hypothesis (1) shows that in a few states, Donald Trump did have more accounts tweeting about him in the second period, but Joe Biden hard roughly the same. The results from hypothesis (2) shows that for both candidates the percent of popular accounts tweeting about the candidates dropped roughly 2% overall. The results for hypothesis (3) shows that for Donald Trump there was a shift of roughly 27.8% net change between negative and positive sentiment while Joe Biden had a net change of roughly 2.3%.

1 2-Epoch Temporal Analysis of Election Related Tweets

1.1 Problem Description

The goal of this project was to carry out several pre-processing techniques on the provided Twitter data as well as on personally gathered data. The topic chosen in this work is for the "Presidential Election". Initially the topic chosen was "Online Games" but after cleaning and beginning the initial analysis, results for the initial hypothesis did not appear or were not interesting. The hypothesis would have required additional information for developed conclusions. The United States Presidential Election is almost upon us and as we get closer to November. The topic was not chosen because of political indecision, involvement, or personal feelings but simply because it is an interesting topic, to say the least. With several crisis emerging everyday such as global catastrophes, stock market inflation, and the current COVID-19 pandemic it is worth trying to understand what and how people discuss, specifically on Twitter, the candidates that could lead the US for the next four years. There are several groups/affiliations/pacts umbrellaed under the two major parties, Democrats Republicans, but without consideration of these an unbiased analysis of what people are tweeting may give some inclination as to what we can expect to see or have seen in the news as the election draws nearer. Much of the knowledge needed for this work comes from the textbook [2].

1.2 Data Gathering

Collection of the second period of tweets was collected following the provided guide "Tutorial on obtaining tweets", therefore the collection of tweets was exactly the same as those in the first time period. Essentially the method for collecting the data is two steps: parse or curate the tweets via the twitter API and then format them from the twitter JavaScript Object Notation (JSON) representation to Comma Separated Values (CSV). Twitter uses the current specification of JSON specification, RFC 8259. To parse or curate the tweets via the twitter API a simple Python script was provided to retrieve the JSON records (per tweet) into a text file that was called "tweets.txt" by running the command `python tweets.py > tweets.txt`. Once the tweets were collected the data was formated to CSV using the provided tweet parser which is another simple Python script that converts each level of each key in a JSON object to a row separated by commmas. The parser was run using the command *python tweetparser.py tweets.txt retweets.csv*.

Specifically in this work the collection of tweets occurred over roughly four hours from 10pm 9/15/2020 to 2am 9/16/2020. An initial collection of tweets was done on August 31, 2020 but the collection process was repeated again on September 15, 2020 to have a more time between the data sets. The keywords used were the same as the keywords used in the provided collection of Presidential Election tweets: "presidential election", "2020 election", "Trump re-election" and "Biden 2020". The size of the data was kept relatively similar to the data set provided with 90,709 records of initial data in the second collection. The intention of the data gathering process in this work was to remain consistent with the previous collection of tweets so that analysis would be adequate.

1.3 Data Exploration

The initial information in the data is difficult to comprehend as there are many levels in the CSV that are created to pivot from an initial JSON object to a CSV. Upon viewing the data set in its essentially raw form there are 71,662 observations (tweets) and 998 columns (features). This very high dimensional set of information is generally filled with null values, in fact not one tweet collected has a value for every feature. This means that every column cannot be used. In the case of this work not every column is needed to answer the hypothesis; however, a deeper understanding would result of having many of these null fields. From a collection point of view several of these fields can be filled if the parser included them. Because it would take to long to investigate the genuine capcaity of each feature to be used, columns were selected with the project hypothesis in mind. This was done by using an Online JSON viewer for two records to consider the contents of what might be relevant fields. Initial findings found that only a small portion of the features would be needed for this project. It is also worth mentioning that even some of these fields have missing data and that different subsets of features were used to analyze and conclude the hypothesis. The initial subset of features that was extracted in this work are *"lang", "created_at", "text", "user.id_str", "user.name", "user.screen_name", "user.created_at", "user.statuses_count", "user.location", "user.verified", "user.followers_count", "user.friends_count", "user.listed_count"*. The dimensionality of the data is now shrunk from almost 1,000 features to 13. These features were believed to have the most impact on formulating a solution to the hypothesis.

From the point of view of noise, *text, user.name, and user.screen_name* could all be considered unstructured data. The features *text and user.location* were noisy and ambiguous to an extent but more attention should be drawn to how noisy *text* or tweet text is and how ambiguous *user.location* or location of the user is. The noise in tweet text exists in the form of misspelled words, imprecise grammar, different languages, humor, among others. The ambiguity in the users location is mostly related to places that cannot be a location (i.e. Listening to Music), not a location that can be considered as usable in the context of this work (i.e The Planet Saturn), or varying degrees of specificity (i.e. Georgia is both a state and a country).

Normalization, a very applicable technique for twitter information, could be applied to several features in the data set such as *"user.statuses_count", "user.followers_count", "user.friends_count", "user.listed_count"*. These features are all numerical and contain values that range from small to very large, both of which are valid in the context of this work. Discretization, another technique that is very applicable here can be applied to specifically,
"user.followers_count" and "user.friends_count" as there are representations in the data that can be gathered if the information is split into meaningful intervals. These are not immediately obvious representations.

There are several features that could be created in this project but two specifically stood out in terms of the hypothesis. The first being hashtags which are a meta tag used within the tweet text that users are able to search, follow, and draw attention to. These would need to be extracted from the text based on the contextual #'s in a tweet text features that could be created in this work are user ratios which are a metric for gauging an individual on twitter. The twitter ratio is explained in very good detail from [ref]. It states that a ratio below 1.0 indicates that you are seeking knowledge and friends but not someone who is heard by many, potentially the account is a bot. A ratio around 1.0 is considered good in the sense that you are roughly being equally heard and equally listening to other individuals. A ratio of 2.0 or higher is a popular individual who is seen or perhaps mentioned in a community. A ratio greater than 10.0 is someone who is popularized by either content or abilities such as politicians, musicians, or a CEO. The ratio is calculated based on the ratio of twitter accounts that follow an individual over the individuals followed twitter accounts.

Additional pre-processing steps that could be needed for the data are cleaning of the tweet text, this includes punctuation, hashtags, artifacts such as links and image embeddings. Removing duplicates is also an important step for keeping each sample unique, this could be based on both the user and the tweet text so that people tweeting similar or "retweeting" a user are still considered as unique. Removal of null information records will be included when it is required. For example the location is not included in every sample so when discussing location a smaller subset of data will be utilized. Sentiment analysis will be used for one of the three hypothesis so an additional column will be created based on the sentiment generated for each tweet text, these values will be *negative, neutral, and positive.* Stop words will not be removed from the text and neither will lemmatization as they have been shown to change the sentiment analysis results [3].

1.4 Hypothesis

The first hypothesis, interest of one candidate will grow more than for the other in some states, asses the geo-location of accounts, specified by *user.location*, to determine if states are tweeting more or less about a candidate. The hypothesis stems from the general expectation that some states generally house a population that subscribes to a specific political party. The expectation is that some states will show more mentions of one candidate over the other in larger magnitude. This would indicate that state may potentially be more interested in a particular candidate at the time of collection. This is kind of weakly correlated but is of interest when considering the states that may or may not need more advertising to increase the chances of a candidate. The number of tweets per state should be shown for both time periods and for both candidates to asses if the hypothesis is correct or not.

The second hypothesis, more popular people will be tweeting about both candidates, aims to deduce popularity of accounts from the features captured in relation to the candidates. Specifically if a candidate is tweeted about by popular accounts in larger proportions. This stems from the conclusion of two major events in political campaigns namely, the DNC and RNC which occurred between the two time periods. People are now more (less) confident in the party candidates so there should be an increase of mentions by popular people, sometimes called endorsements. This will be done by considering several factors related to the accounts numerical data. The twitter ratio will need to be calculated and considered when describing popularity. If there is a meaningful threshold within the data that appears a ratio of 10.0 or larger has been cited as meaningfully popular and will be used [1]. The ratio will determine how influential the account is when tweeting about a candidate. The ratios below 10.0 would not make sense to consider unless they show some other characteristic stating high popularity or influence. Because of this, follower count will also be used to distinguish popular accounts as people with 100,000 followers might also be considered popular. The verified status of an account will also be considered as these are people considered by Twitter to be individuals with a "public interest". This metric could potentially be the best metric for answering this hypothesis.

The third hypothesis,sentiment will shift in either positive or negative directions for oneof the candidates by 5% or more, will seek to discover if the sentiment per candidate has changed overall. The shift from negative to possitive or vice versa about one candidate could mean people are more in favor of approving a candidate. We can assume that this arises from an inital degree of sentiment changing. For example, if tweets related to Joe Biden and Donald Trump are 70% 80% positive respectively, in the first time period, but they are drop to 68% and 30% then the change in sentiment behavior would suggest a large negative displacement for Trump and a small one for Biden. More over if there is a dramatic change in sentiment that could mean a more charged environment most likely due to differences of opinion or news shedding light on a candidates history. The tweet text will be used to ascertain this information. Specifically sentiment analysis will be performed on the tweets and a comparison between the two will be conducted.

1.5 Preprocessing

1.5.1 Software Tools

To preform pre-processing on the data Python3 was used, the specific libraries were regex (re), numpy, us, pandas, and TextBlob. Regex or regular expressions were used primarily for the tweet text. These were used to extract or remove information. The numpy library was used to preform some simple matrix operations. US is used for united states cities and states. Pandas was used for dataframes as dataframes are the typical format for pre-processing information in modern programming languages. TextBlob was used for generating the sentiment analysis. I used Jupyter notebook and Visual Studio Code for writing the code and work was done on a Windows OS.

1.5.2 Techniques

Tweets were initially filtered based on the language code "en" or English. This was done for consistency as other languages were not understood. Filtering by language code en still retains 96.68% of the data so this did not cost to much in terms of losing data. The tweet counts for other language codes is show in 1. These values show that the tweets with other language codes are almost outliers in the data set when considered separately. They could be aggregated into two separate categories for "en" and "other language" but again, we would not be able to understand every tweet. It should be mentioned that the data in this section only shows information from the initial data set but each manipulation and technique used was applied to both data sets.

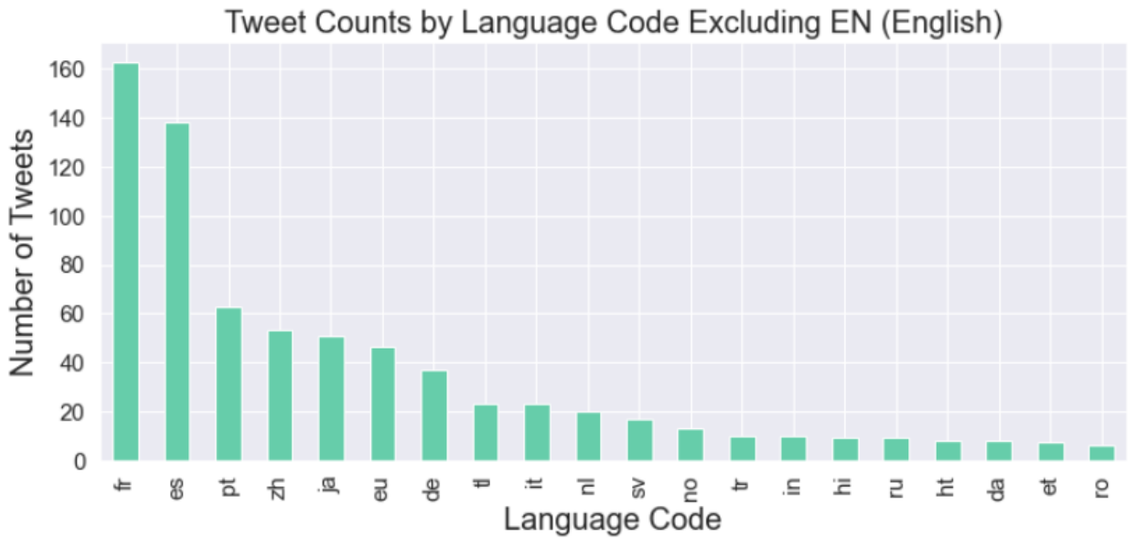

Figure 1: The number of tweets by language code excluding en or "English". the small number tweets shows they are essentially outliers to the data set

Sentiment analysis was used on the tweet text to ascertain the sentimental value for each tweet. The sentiment of a text can be found by using a function from TextBlob that will automatically calculate the sentiment score among additional information about a piece of text. Sentiment analysis is a large field of study for purposes in this work the sentiment of a tweet is discretized into three sentimental values, "Negative", "Neutral", "Positive". These values were founded based on the ranges documented for TextBlob sentiment. Values less than 0.0 are considered negative, values equal to 0.0 are considered neutral, and values greater than 0.0 are considered positive. The values were created for each record and then discretized. For each tweet this was calculated using simple function that takes the tweet text from a data frame row, cleans it of any artifacts such as links, embeddings, and special characters. The TextBlob class then takes the cleaned text and computes a sentiment score that can range from [-1.0, 1.0]. The discretization was performed by setting the conditions for *less than 0.0, equal to 0.0, and greater than 0.0.* In a sense performing sentiment analysis on the original data was a pre-processing technique as we create the information from the contents of the data. Discretization, another technique is also used here to separate the sentiment into meaningful ranges. The values could be normalized over the sample space using standard deviation normalization. The ranges could be created by evaluating a certain portion near the median after the normalization. But since TextBlob has already normalized the values between [-1.0, 1.0] it did not make sense to normalize but to discretize instead. The ranges were used following the functions documentation, this is mainly because we could subdivide the ranges of negative, neutral, and positive infinitely using as many prefixes as we like: Ultra, Extremely, Very, Partially, Almost Negative/Neutral/Positive. The sentiment will be considered for each candidate by looking at the records with mentionds of the candidates for both time periods.

An additional field was created to incorporate the information in followers and friends. Referred to as the twitter ratio or ratio in the context of this work, it is a simple division of the number of followers an account posses over the number of accounts the account follows (friends) it is calculated as follows in Equation 1:

$$twitter\_ratio = \frac{account\_followers}{account\_friends} \tag{1}$$

where a high ratio is typically hard to achieve and indicates popularity/fame. The data has 51,039 or 74% of the accounts with a ratio value of less than 1.0. The Figure 2 shows the ratios for accounts under 2.0 with. They show a rather convincing argument for accounts largely below 1.0 while there are many accounts that are trying to maintain a ratio of about 1.0.

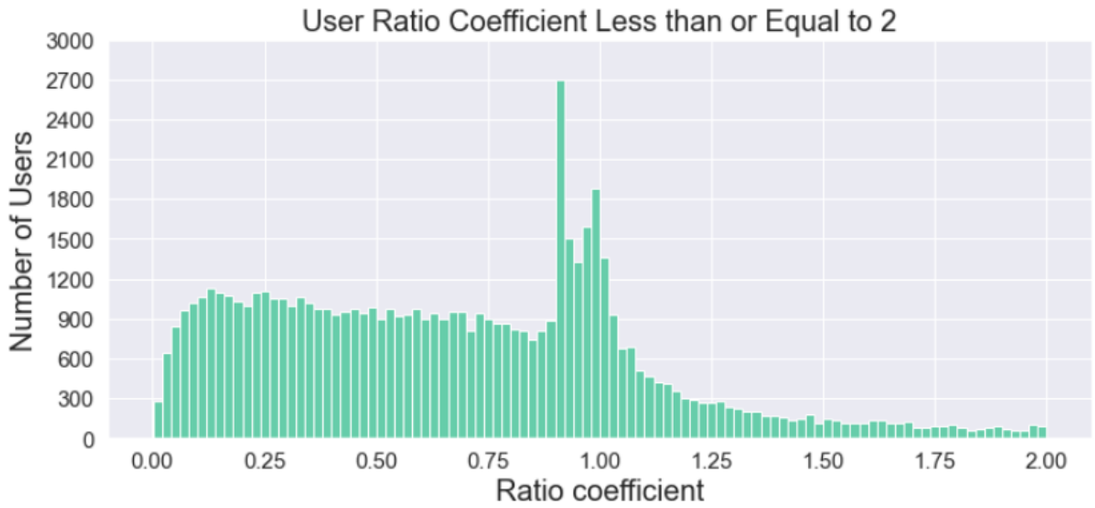

Figure 2: The ratios of users with a value less than or equal to 2.0, noting the spike at 1.0 and the skew right in favor of values less than 1.0.

The ratios for values greater than 2.0 but less than 20.0 are show in Figure 3 as they show a dramatic drop off in account ratios. The ratio values in this figure show that fewer and fewer users have a high ratio value because it is hard to maintain without popularity or fame.

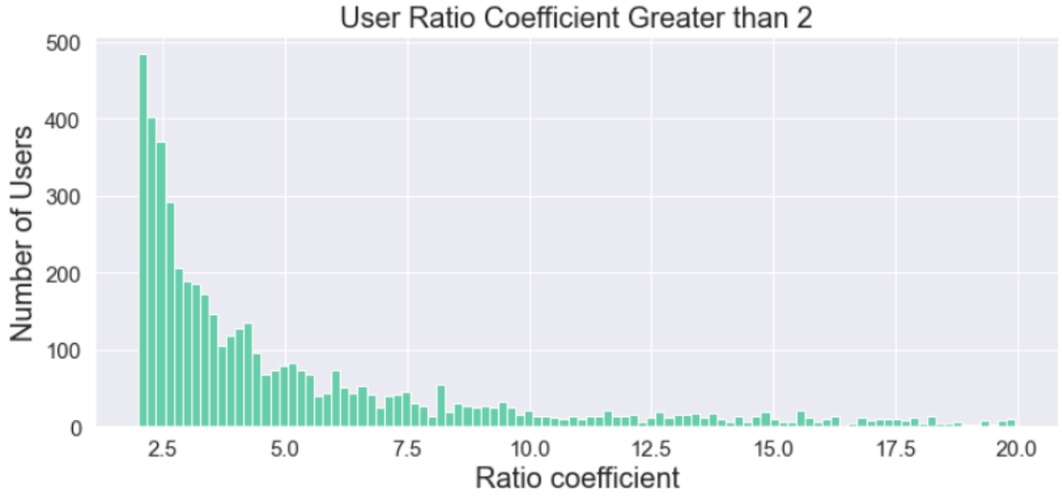

Figure 3: The ratios of users with a value greater than 2.0, noting the drop off in high number of users with increasing ratio size.

Essentially the ratio shows that popular people, people with large ratios, are few and far between in comparison to the vast amount of users. The ratio coefficient can be any number that is not greater than the number of followers a user has. For example, if a user has 10,000,000 followers and only follows 10 people their ratio would be 1,000,000. In total users with a ratio greater than 10.0 is 593 or 0.864% . Because of this large difference values standard deviation normalization was used to lower the breadth of complexity in this feature space but the values being so variable needed a representation that still showed the large deviations between a ratio of 0.4 and 27,000. The original maximum and minimum values for ratio were calculated as 59,437 and 0.0007, respectively. After normalization the minimum and maximum became 185.331 and -0.263, respectively. Ultimately the normalization of the column could not be used as popularity was based on the 10.0 ratio found in Follower-to-Friend Ratio for Steemit, but it did give insight into how the data could be better understood or used in a different application.

After evaluating the ratios individually the ratios were also considered with the verified status of each sample's account. The verified status may or may not be the best indicator of a sample's popularity but it does indicate that an account has gone through the entire process of account verification which has some guidelines/restrictions. When considered with the twitter ratio created it shows that proportionally verified accounts have higher ratios than non-verified accounts. That said, not all popular people are verified on twitter have a high ratio of followers to friends. Visually it is very difficult to see as the distribution of peaks in the latter half of both plots in Figure 4 appears in favor of verified accounts. Numerically, the percent of verified accounts with a ratio higher than 10.0 is 61.1% while for non-verified accounts this is about 31.1%. In both cases for ratio we consider a high ratio to be 10.0 because at a ratio of 10.0 it becomes rather apparent who is very popular and who is not for someone who has a million follower to have a ratio of 10.0 they would need to be friends with 100,000 people on twitter. This may seem intractable and that is in general the rational. With many followers this ratio is not hard to achieve. An account with 20,000

followers would need to have less 2,000 friends which is significantly harder to maintain. With that rational, the 61.1% of users mentioned could have any number of follower as long as their ratio is higher than 10.0. What was found is that there are many accounts that have a ratios higher than 10.0 but only have 2,000 or even 500 followers. This kind of popularity is not really what was aimed for. We are looking for popularity that reaches larger volumes of people. In the final evaluation people with more than 5,000 followers were considered with ratios higher than 10.0.

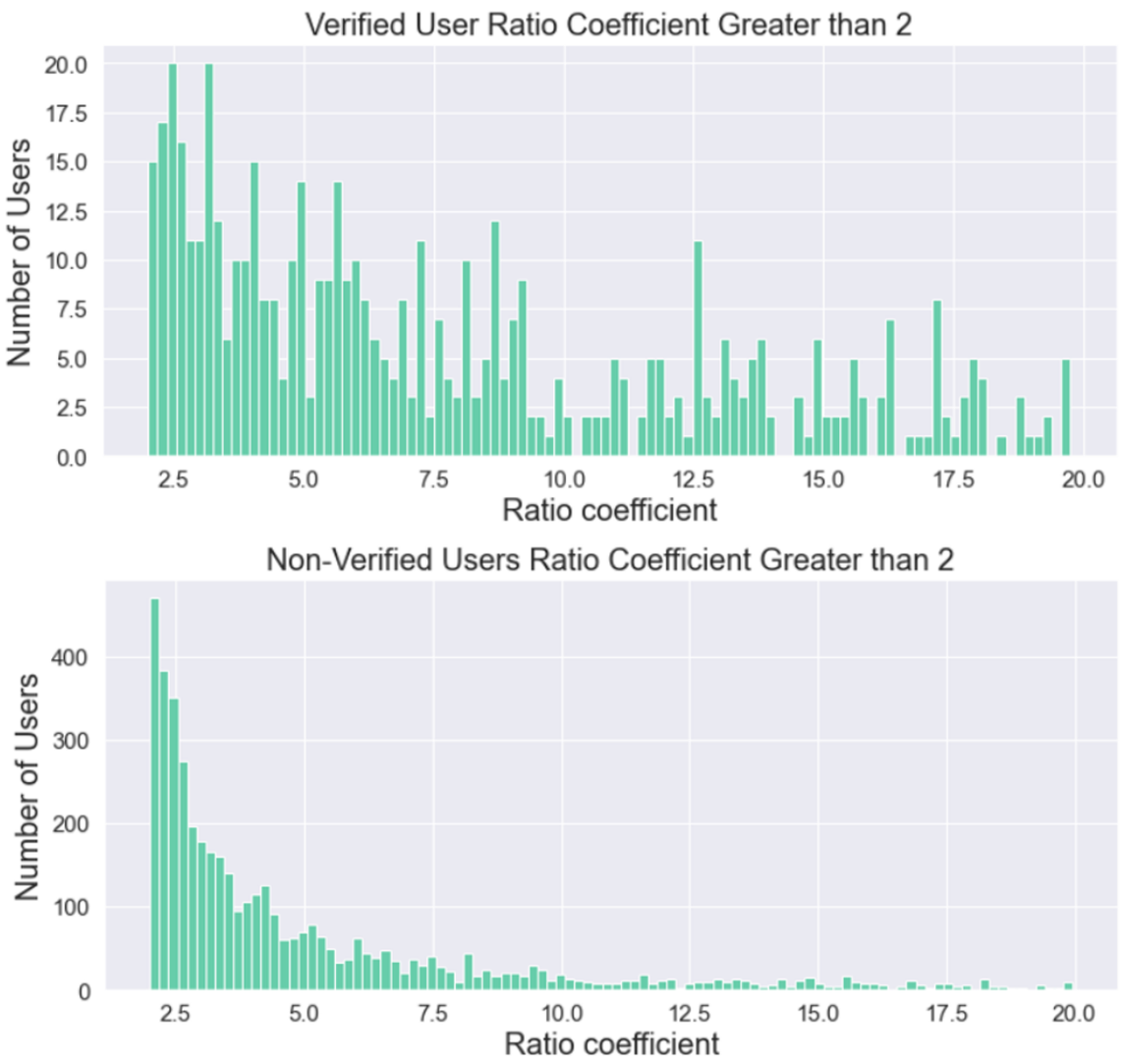

Figure 4: Verified users after normalization showing that ratio does not signal verification. But proportionally higher ratios are more significant in verified accounts.

In addition to ratios, and in the context of influential or popular accounts, we do not want to exclude individuals who have a large twitter following. People who fit this description but do not have high ratios maybe communities or groups that do not care about maintaining a ratio and are more interested in keeping their members. One of these examples, specifically in this data set is the account *@cvheady007* who has more than 282,000 followers but also more than 310,000 accounts that he follows. If we were solely basing the analysis on ratios this account would not make it with such a low score as 0.91, without normalization. However, followers also are a claim to popularity especially when an individual can directly reach over 200,000 people with one tweet. To be fair we are not considering the falsified influence a user might have. For example, many accounts may have bot followers to increase their awareness or ability to interact with people on a broader scale. The distributions in 5 are of four different portions of the data set. They were discretized initially and eventually only the last distribution was of use in answering the related hypothesis. It should also be mentioned that most accounts are locked to 5,000 friends. Looking at the third plot in Figure 5 we can see a accounts clustering around where 5,000 friends is located. A simple regression was used to plot a fitted line to the distributed points. This was not used for any further analysis other than visualization. It appears that the data is approaching a 1 to 1 ratio of followers to friends based on the regressed line.

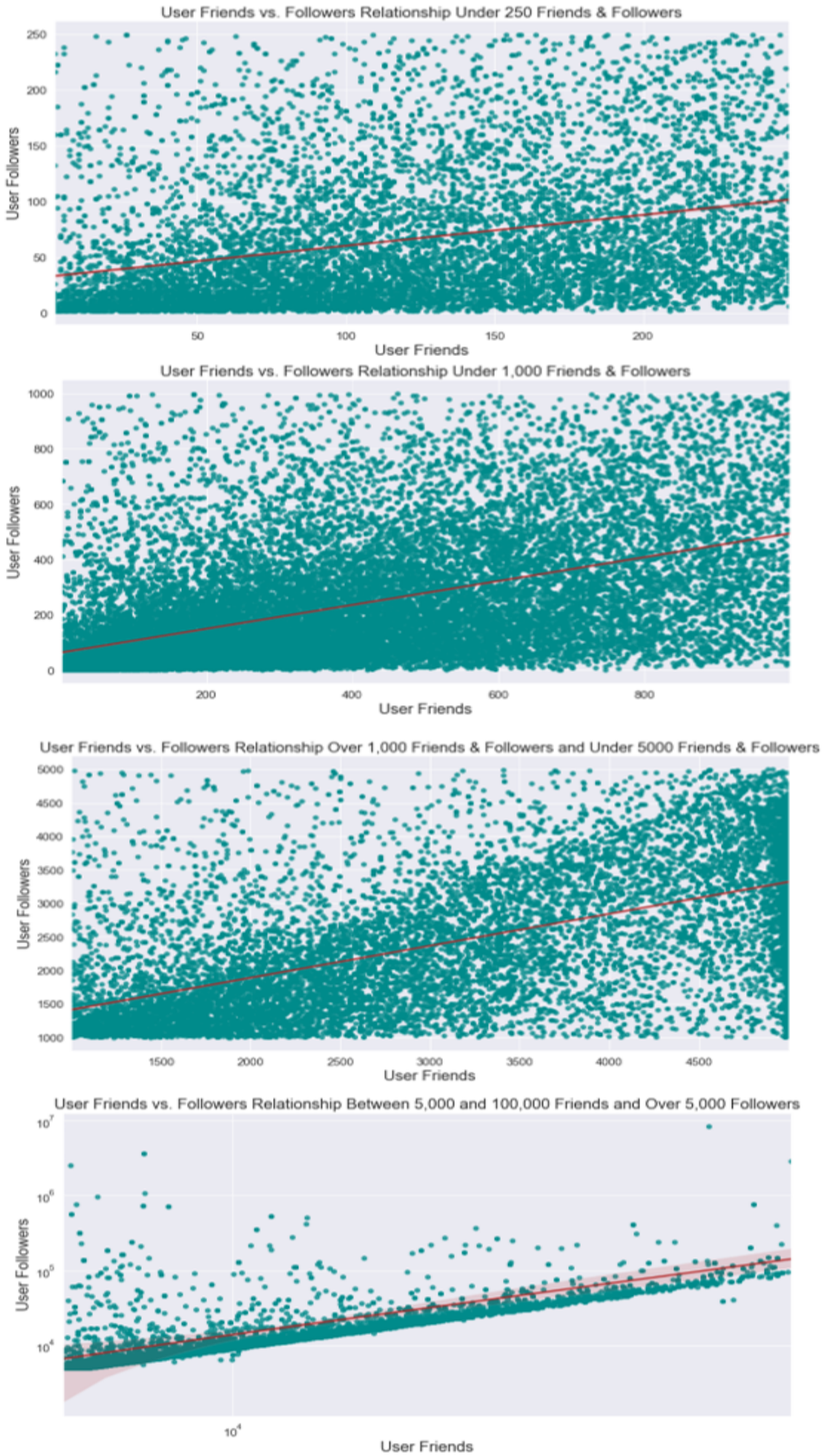

Figure 5: From top to bottom the distributions of friends and followers of accounts in the data set with increasing range values from as little as 250 followers and friends to millions of followers and 100,000 friends. The last of these was plotted using a log scale because of the high variability between the data.

Hashtags are a meta tag that users are able to use in order to label, search, and highlight information that is, but not always, included in the body of a tweet. The format of a hashtag is *#[sometext]* where sometext can be any string of text not separated by spaces. To extract hashtags a regular expression was used to extract each of the hashtags in a tweet's text. For example, the text "Joe Biden picked a great candidate Kamala I hope the best for them. Joe2020 Kala2020 BidenHarris", has four hashtags. When cleaned a column will be created and should contain a list of hashtags like ["Kamala", "Joe2020", "Kmala2020", "BidenHarris"] regardless of any misspelling, such as in Kamala2020. Each tweet was processed with a simple function that takes a tweet text, applies the regular expression to find and extract the values following hastages and then a pipe separated string is created in the example above this would be "Kamala—Joe2020—kmala2020—BidenHarris". The pipe separated string was created so that it can be both searched or separated easily. In total the number of tweets that have hashtags was relatively low, 6,071 tweets, accounting for about 8.85% of the tweets. Hashtags are usually used to determine trends and popular topics within the twitter environment. Although they may be misspelled or incorrectly used they should give more incite into a candidates popularity. It was found that users tweeting about Donald Trump use less hashtags than users tweeting about Joe Biden, this could lead to more interest, in terms of a trend, or recommendations by twitter to show users Joe Biden related content, because based on hashtags he is more popular but overall Trump had significantly more mentions. The opposite is true for the candidates vice president picks, Mike Pence had almost no hashtag mentions while Kamala Harris had roughly the same as Trump. In the second data set Biden had both more hashtags mentions and overall mentions then Trump and Harris still had significantly more mentions in both categories than Pence. Ultimately with just a few thousand records for each candidate this was not used to evaluate popularity.

For cities the data begins to get rather sparse in comparison to the other data that was created with pre-processing as not everyone lists their actual location, or any location for that matter. Many people have some metaphor or incorrect location for their user location. For example a user could live in the city Jupiter Florida, but when specifies their location is "The Planet Jupiter" it becomes difficult to separate these differences without complex methods, it may not even be possible in some scenarios. The text for each location was cleaned so that no special characters would remain. The package 'us' was used to then lookup if any of the abbreviations for states or state names were included in the body of the user location feature. A function was created to abstract this such that if the result was a state or state abbreviation the result will be used otherwise the data is not included. The comparison is for specifically US states so there is no way to include accounts that list just a city or just USA. The states were relatively diverse although there is skewing in the direction of states with high populations as can be seen in Figure 6 this is something to be expected as California, Texas, and Florida house a sizable amount of the US population. The state geographical plots are also included to show the heat map of tweets for each state. The heat map is a good representation when the data can be easily coerced into states and simplify the state by state relevance based on the color gradient. In Figure 6 the states with more tweets are more yellow and those with less are a deeper blue. The bar chart was included additionally as it is also a clear representation state by state, listing individual state names and state values.

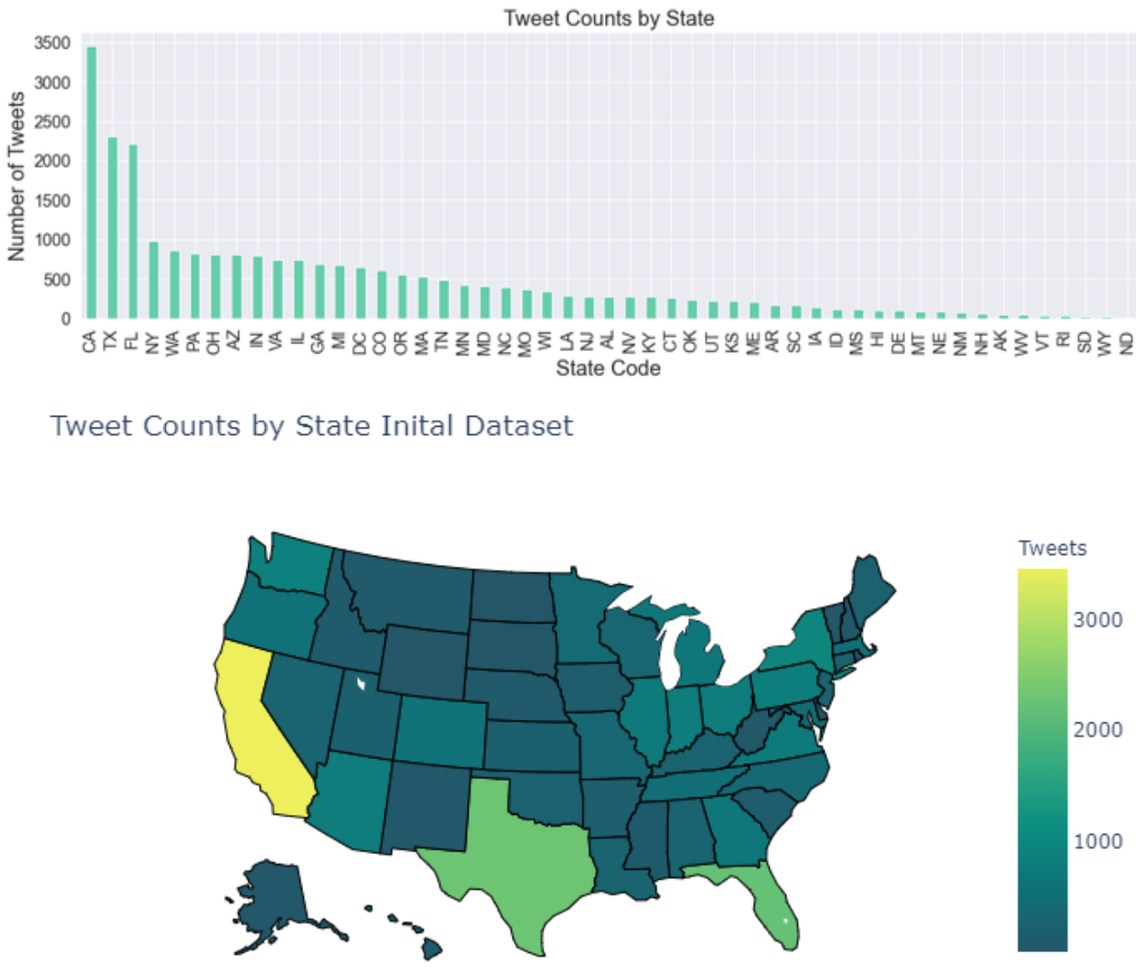

Figure 6: The number of tweets for each state. Plotted both in bar plot and spatially mapped across the united states for both the initial data set.

Aside from these very specific pre-processing methods. Samples from the data set were removed in general based on three assumptions: records that do not have data in all of fields *"lang", "text", "user.ratio", "user.friends_count", "user.listed_count", "user.verified"* will be removed (loss of 200 records), records with a ratio less than 0.000 are removed (loss of 300 records), records that have the same *"text", "user.id_str", and "user.screen_name"* are removed (loss of 1,100 records). The first of these filters was done to remove records that cannot be used in the majority of this analysis. These were removed to keep the data set homogeneous when performing the rest of the pre-processing steps. The second removes people who only follow other users. These accounts could very likely be bots, but more important their tweets are not seen by group of people. Tweets are available for anyone to search in general but the influence or existence of someone with no friends on twitter is essentially not worth using in this work. The third step removes duplicates from the data set. Duplicate tweets are not the same as duplicate samples. Several accounts may tweet the same information and that would have some influence over the data set; however, records that are from the same user with the same text are not new observations they are for whatever reason included. This maybe due to some feature of the initial twitter collection process.

1.6 Visualizing Patterns

1.6.1 Software Tool

To create visualizations from the data work was done in Python3, the specific libraries used are Matplotlib, Seaborn, and Plotly. Matplotlib was initially created by a neurobiologist who who wanted better plotting methods during his doctoral thesis. It is the foundation for most of Python's plotting abilities and is comprehensive in allowing a user to create new and interesting visualizations. Seaborn is a library which is written on top of Matplotlib and makes some of the plotting clearer and nicer without having to individually do so by hand. It also is very well integrated for use with pandas Data Frames which were used in this work. The third plotting packaged used is Ploty which has many abstractions not implemented in Matplotlib or Seaborn, many of which are based on HTML markup. Specifically the heat map of the US was created using Plotly.

1.6.2 Visualizations

Visualizations used over the data set were bar plots, histograms, scatter plots, pie charts, and the GeoMap for the states. Bin size for all histograms was set to 100 because of the size of the data set being over 50,000 records. The plots were used for both the initial data set and the second data set. The shift or temporal aspect due to some time change between the initial data set and the second is compared here.

The GeoMap used shows very clearly that a few states are tweeting more about different candidates while many are relatively the same. It also shows that some states like Texas and Florida grew in favor of both candidates almost equally. Figure 7 shows both collections of tweets for each candidate. The gray/whitish states are states that did not have data. Each state color represents the number of tweets for that state, specifically for each candidate. It shows that generally many of the states are similarly distributed when discussing the candidates. Based on this visualization, yes hypothesis (1) is correct but it is not as dramatic as expected.

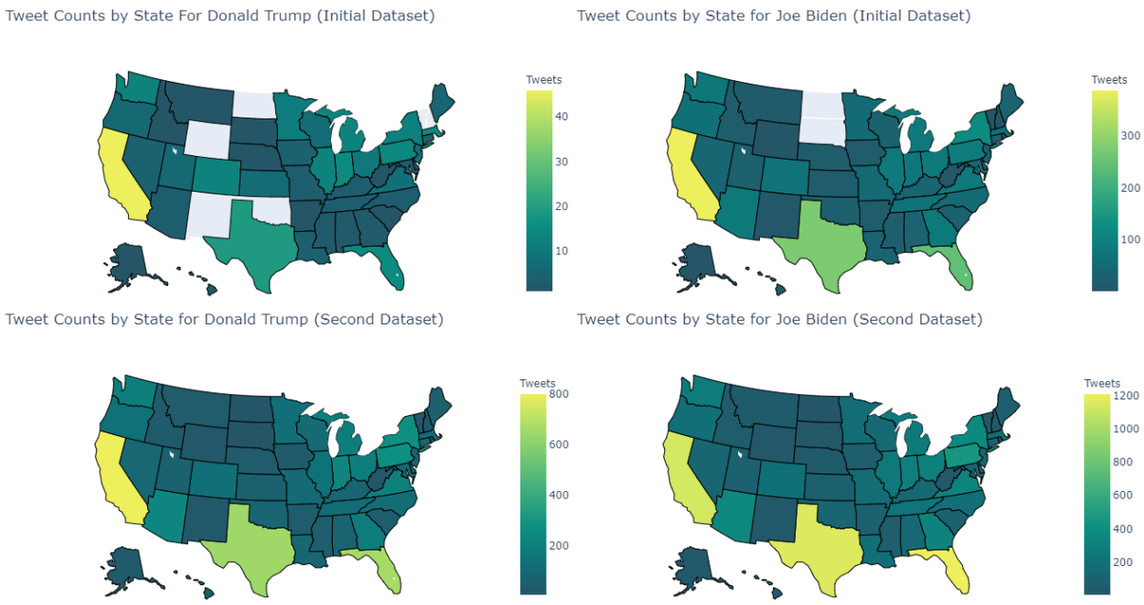

Figure 7: The number of tweets for each state. Spatially mapped across the united states. From top to bottom, initial and second collections of tweets. Left to right, candidates Donald Trump and Joe Biden.

Figure 8 shows the popularity comparison of tweets per candidate using a bar chart. The bar chart was used because the percentages are close, the data is not complex (there are only two features), and it represents the difference clearly. It was based on the number of tweets that high ratio accounts with more than 5,000 followers had for each candidate. It shows a drop in the number of tweets which is interesting as it was expected that as the election gets closer the numbers would increase. Based on this visualization, no hypothesis (2) is incorrect.

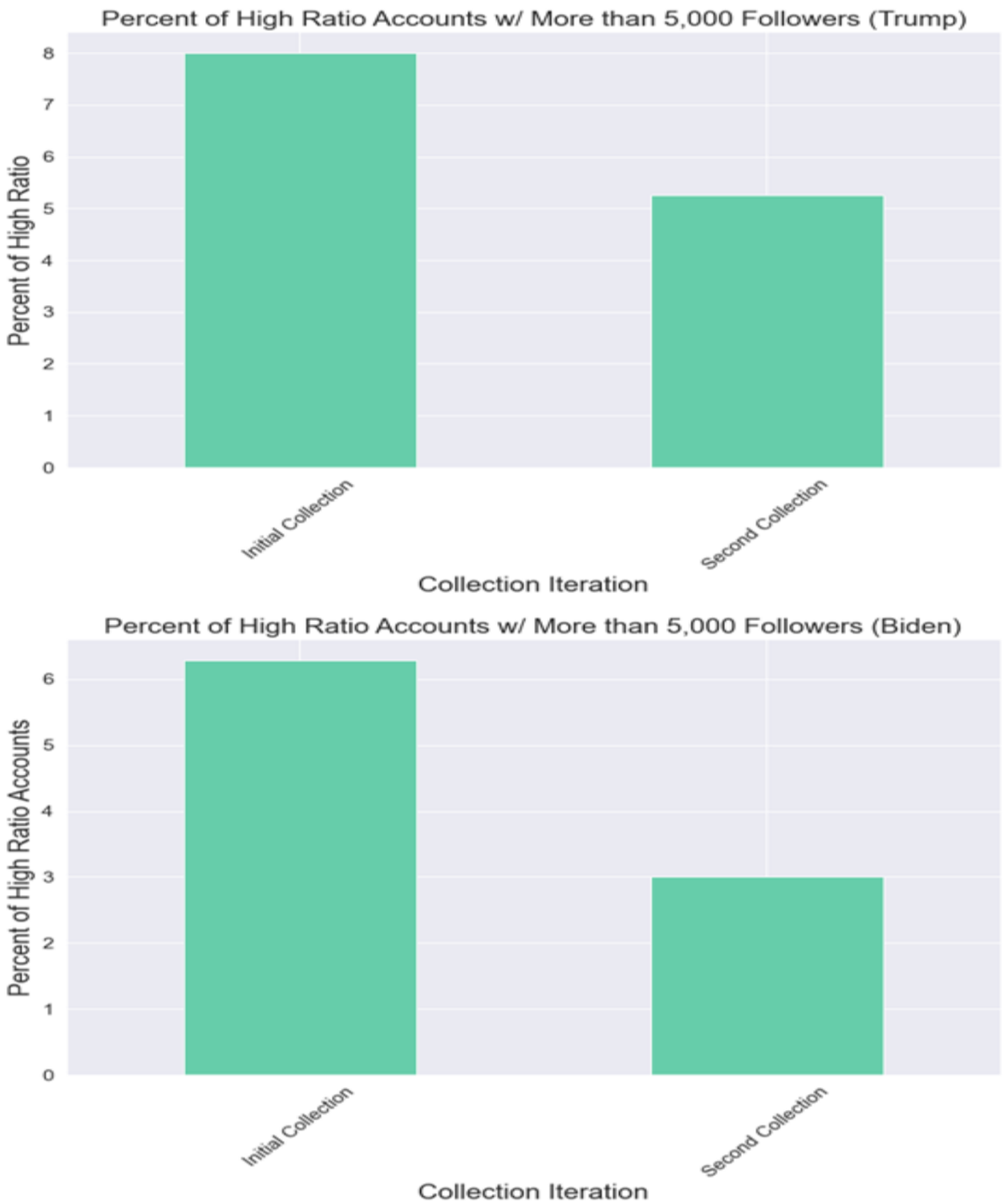

Figure 8: The percent of tweets for each candidate from high ratio accounts with more than 5,000 followers. From top to bottom, candidates Trump and Biden percentages.

Figure 9 shows the sentimental comparison of tweets per candidate using a pie chart. The pie chart was used because the second data set collected had more than 5,000 and 6,000, or roughly 16% and 20%, more records for Donald Trump and Joe Biden respectively. Each portion is colored based on the proportion of tweets that were negative, neutral, or positive. The change over time is easily apparent after using this visualization. Based on this visualization, yes hypothesis (3) was correct.

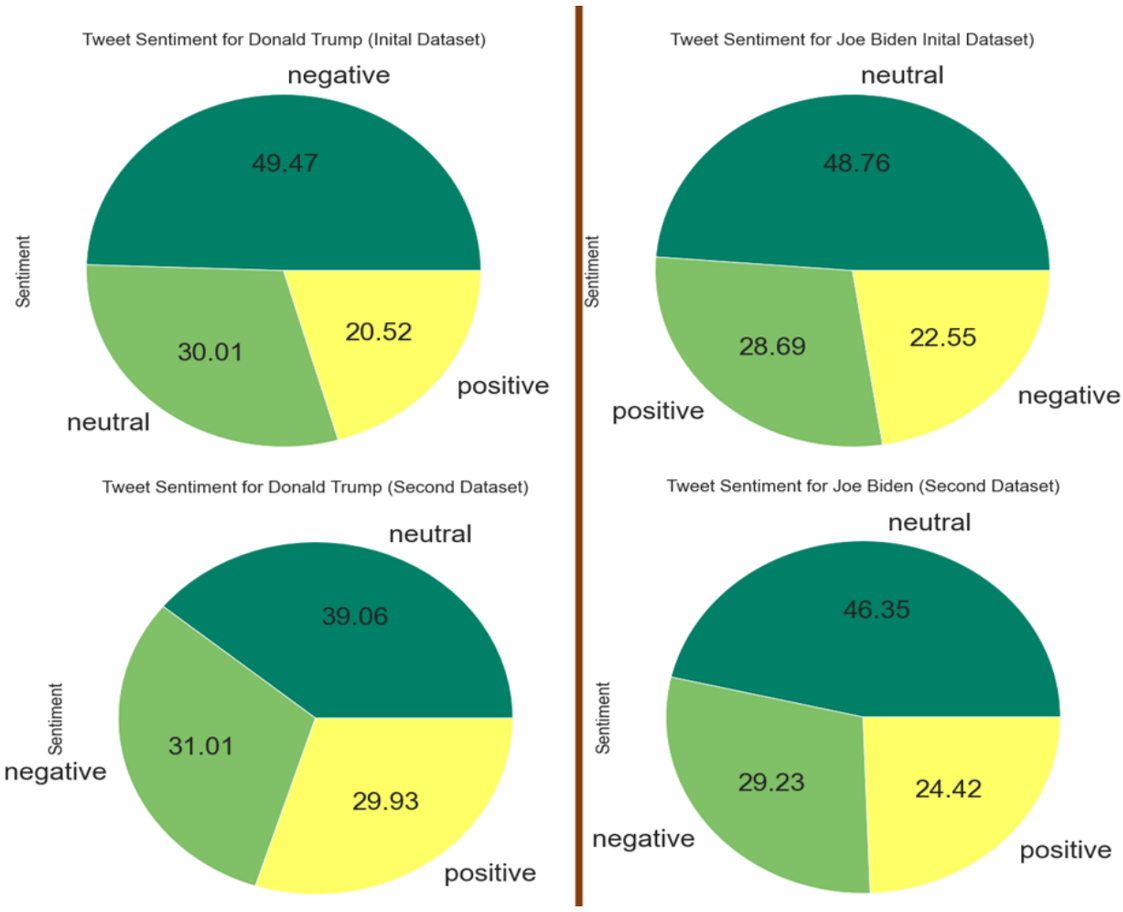

Figure 9: The sentiment of tweets for candidate. From top to bottom, initial and second collections of tweets. Left to right, candidates Donald Trump and Joe Biden.

1.7 Conclusion

Hypothesis (1) seems to be correct or in a very small sense it could be considered correct. The states that were not tweeting about Donald Trump, New Mexico, Oklahoma, Wyoming, and North Dakota, in the initial data set do tweet about him in the second set, which is more and with pretty much the same consistency as the others low to medium volume states. The results were a little bit different from what was initially expected. Based on recent news things are actually stable, this is based on polling statistics, Joe Biden has been leading with consistently the same number of points over the last six months. It is apparent that different states having slightly more tweets about one candidate in the second sample are not that different from the rest of the states. Excluding the high population states like California, Florida, and Texas the states were almost unchanged in their number of tweets.

Hypothesis (2) appears to be clearly false as the percent of high ratio accounts tweeting about both candidates decreased. This decrease is interesting as the election is less than two months away. Endorsements typically happen over and over again, not just from popular accounts but in the real world people are often ready to "endorse" people regularly. The decrease could be due to the period in time we are currently in. As we draw closer to the final 1-3 weeks of the campaign these percentages may spike. Additionally the DNC and RNC occured several weeks ago which is when most endorsements are shared publicly. The drop in percentage could be due to the small drop in interest as we are about to ramp up to maximum interest in the election over the next several weeks.

Hypothesis (3) appears to be completely true for Donald Trump but not true for Joe Biden. The hypothesis was for only one candidate so this hypothesis was correct. The hypothesis that a change in sentiment would occur was mainly based on the politically charged atmosphere that comes with campaigns. The actual assumption was that when evaluting the difference for both, each candidate would become more negative. This was because of the recent new and state of America with many people falling on harsher times from economic failings (in small buisnesses), protests, the COVID-19 pandemic, climate change, among other things. The large increase in positive tweets coming from Trump related tweets could be several things. Sentiment analysis can be difficult to correctly determine causality. The assumption is that people are satisifed with how the president is doing things. The slight change in sentiment for Joe Biden is nominal therefore it is not really considered as a change. The real insight gained form this hypothesis is that in some cases sentiment can spike for a candidate and in others it may be harder to change.

The difficulty with these hypothesises, is that there are only two epochs of time. The data could have been split every hour, or every quarter hour to get a more time distributed temporal analysis with many more epochs. The reason this was not done was because the analysis in this work can be applied over any number of epochs. The inital inferences needed to be made about distributions and overal mentions, hashtag use, etc. It would be interesting to apply the work here to several time periods and compare the anylsis.

References

[1] jfollas. A "follower-to-friend" ratio for steemit? 2017.

[2] Mehmed Kantardzic. *Data mining: concepts, models, methods, and algorithms*. John Wiley & Sons, 2011.

[3] Hassan Saif, Miriam Fernández, Yulan He, and Harith Alani. On stopwords, filtering and data sparsity for sentiment analysis of twitter. 2014.