Journal of High School Science

Sentiment Analysis on Tweets in the 2020 US Presidential Election

Chandak A

Submitted: December 5, 2021, revised version 1 submitted January 4, 2022, revised version 2 submitted January 16, 2022
Accepted: February 4, 2022

Abstract

Social media played a significant role as a medium for advancing, influencing or promulgating specific political motives, in the outcome of the 2020 US Presidential election and in the events following it. Social media provided a way for candidates to reach their followers and voters and share their views and opinions on particular topics and vice-versa. This paper analyzed several sentiment analysis models to examine the correlation between Twitter sentiment trends and election results. The greatest accuracy was obtained by using the Valence Aware Dictionary and sEntiment Reasoner (VADER) python library. Twitter sentiment analysis can be used to improve campaign effectiveness and to predict the outcome of elections.

Keywords

Sentiment Analysis, Twitter, Election, Natural Language Processing, Social Media

Corresponding Author: Armaan Chandak, armaan.chandak@gmail.com. Mary Institute and St. Louis Country Day School, 101 N Warson Rd, St. Louis, MO 63124.

Introduction and Summary

The recent US election has been the cause for increasing division and controversy between political parties. The dynamics which incited political devotion and loyalty also helped encourage radicalism and violence. There are countless examples of modern political hysteria, including the recent capitol riots, that occurred after the election. Furthermore, the emergence of social media, specifically Twitter, has added another dimension to campaign strategies because of its ability to simultaneously reach a large population and give the ordinary person a voice (1). The increasing use of social media has also played a significant role in political party allegiance and election outcomes (2).

In this paper, I analyze three Python sentiment analysis models, a Bag-of-Words model, a model trained on sentence embedding, and two Out-of-box classifiers to predict the sentiment of election-related tweets. The Out-of-box classifiers, specifically the Valence Aware Dictionary and sEntiment Reasoner (VADER) library, provided the highest accuracy. The results of the sentiment analysis models were a set of tweets classified as either positive, negative, or neutral and sorted by the date created and with the associated political party. Based on the sentiment classification models, the overall sentiment of the election tweets was determined to be more negative for Trump and more positive for Biden. Furthermore, after polling, before the results, Trump tweets became more positive, and Biden tweets grew negative. As Biden approached victory, the trend flipped and Biden spiked in positive tweets; Trump, in the negative. It was concluded that the candidate determined the sentiment trend, and voters followed, showing the power that one can hold over a group of people with the aid of social media to amplify preexisting beliefs. The sentiment of tweets could develop as a crucial tool in future election campaigns in understanding ordinary peoples' thinking and in predicting the outcome of elections.

Datasets

The two datasets I used for this task were Sentiment140 and US Election 2020 Tweets (3, 4).

The training data, Sentiment140, is a popular and reliable sentiment analysis training dataset. It contains 1,600,000 tweets that were automatically classified without manually annotating them. Their model tests three machine learning algorithms: Naive Bayes, Maximum Entropy, and Support Vector Machine. Their feature extractors are unigrams, bigrams, unigrams and bigrams, and unigrams with part of speech tags. This framework treats

feature extractors and machine learning classifiers as two different components to be able to try different combinations of them. These tweets were examined for their emoticons (textual expression representing facial expressions) such as ':)' and ':(' in order to classify them as either positive or negative.

The dataset of election tweets was obtained from the Twitter API. Each tweet includes election-related keywords and was created between October 14, 2020, and November 9, 2020. The time frame of tweets extends after the ballots were cast on November 3, which captured the voters' reactions while the election results were coming in. These tweets were separated into Trump-related tweets using #DonaldTrump and #Trump keywords and Biden-related tweets using #JoeBiden and #Biden keywords. However, it is essential to note that the candidates' tweets were not necessarily written by supporters, therefore showing the overall sentiment towards the candidate from either side. Finally, there were 970919 tweets in the Trump dataset and 775054 tweets in the Biden dataset.

In order to generalize the training data for use on the test data, it is important to examine the similarities and differences between the two. Tweets use informal language and irregular words, so both datasets must be collected from tweets. However, the training data is generalized for all types of tweets, whereas the election tweets solely contain political content from a shorter duration of time. Since the election tweet dataset was collected in 2020 and the Sentiment140 training dataset was collected in 2009, there were minor changes in the language style; however, the difference in language did not necessarily interfere with the model's ability to understand these tweets. Therefore, the training and test data are similar enough that training data can be generalized for the test data. Finally, it is essential to note that each tweet from the Sentiment140 dataset was labeled positive or negative; however, the election tweets were positive, negative, or neutral.

Although these datasets are open-source and taken from Kaggle, a community-maintained platform, the ethics of using these datasets must be acknowledged. In a study of ethical data use, there are no norms or guidelines that have been explicitly agreed upon (5). However, many individuals present different beliefs on what types of data can be used. The use of tweets in these two datasets is designed to be unbiased, meaning that the users do not know their tweets are used for analysis. Although Twitter users do not give consent to data collectors, let alone know that their tweets are being used for research purposes, Twitter's Privacy Policy was updated in 2014 to allow researchers to use their tweets. Hence, it is legal to use these

datasets; however, it is still essential to discuss the ethics of using them (6).

Methods

I used three sentiment classification methods to determine the polarity of tweets in the Election Dataset. With each model, the tweets were assigned as being either negative, neutral, or positive. The method used to define a tweet as negative, neutral, or positive follows in the next section. Although measuring the intensity scores for each tweet could have been beneficial, intensity scores are not provided in either dataset. Determining the intensity scores by calculating the number of times each feature is used in a positive or negative context provides inaccurate data since intensity cannot be measured by frequency alone. For example, "not good" will be classified as being negative with the same intensity as "horrible," however, they obviously have different intensities. Neutral tweets had to be included in classifications because some tweets did not include enough text to be positive or negative, or they posed a general statement, not regarding a candidate. It was necessary to include neutral tweets since they can also determine trends in the data, such as the intensity of polarization. An explanation of how these tweets were classified follows in the next section.

Self-Coding and Codebook

Since the Election Tweets dataset did not include sentiment scores, I self-coded 110 randomly chosen tweets for Biden and Trump (220 total) to compare with the models' evaluations. The number of tweets was reduced to exclude tweets that would have been removed from training data (different languages, no text). I also created a codebook after examining different tweets that were used to self-classify other tweets. A code is a label to tag a concept or value found in text; codebooks consist of a compilation of these codes to help evaluate the sentiment of the tweet. For the election, positive tweets are generally promoting or supporting a candidate, something that encourages people to vote for them; negative tweets generally have the intent of discouraging people from voting for a particular candidate. Neutral tweets do not promote or discourage voting for either candidate. Some of these codes were simple text relationships such as "Hunter Biden" and "censorship" classified as negative or "excited" and "wise" as positive. However, other codes covered ideas and concepts. For example, tweets encouraging voters to vote were almost always positive such as "Vote for #JoeBiden" or "I voted for #DonaldTrump". Furthermore, tweets questioning the regulations or qualifications of candidates were almost always negative such as "Was it really worth the

sacrifice of the thousands of people to Covid just for the benefit of the stock market??". Questions like "When is the election happening?" which were not directed at any presidential candidate, were deemed neutral, and sarcastic questions or comments were deemed negative. The self-coded Trump data contained 27 positive tweets, 11 neutral tweets, and 52 negative tweets. The Biden data contained 31 positive tweets, 16 neutral tweets, and 24 negative tweets. The randomly selected tweets were used to validate and improve the model's accuracy; the final model was uniformly applied to both Trump and Biden thereby eliminating bias between candidates.

Bag-of-words

The Natural Language Toolkit (NLTK) is an important Natural Language Processing library with various features, including classification, tokenization, and tagging (7). A feature is a word or set of words. The bag-of-words process converts text input into vectors based on how many times a feature appears. The process involves multiple steps: cleaning, creating the feature matrix, and training the model.

Cleaning and Preprocessing

The first step in a bag-of-words model is cleaning and pre-processing to turn raw data into a useful form. I applied a variety of cleaning processes. Initially, I removed non-ascii characters, extra spaces, links and HTML tags such as "&amp". Next, using Langdetect's detect function, I removed non-English sentences from both datasets. Finally, I used NLTK's word-tokenizer, a method that splits strings into tokens to remove stopwords such as "a", "is", "the" that do not affect the meaning of the sentence. After testing different cleaning methods, a higher accuracy was obtained when removing mentions that follow symbols and separating hashtags like "#VoteDonaldTrump" into three separate words "Vote Donald Trump" based on capital letters in the tweet.

Feature Matrix and Training

Next, I transformed the set of tweets into a feature matrix, a process that counts the number of occurrences for each unique n-gram, a word or series of words, using NLTK's CountVectorizer. Finally, using the Scikit-learn library, I implemented three machine learning classification models, LogisticRegression, SVM, and RandomForest, to determine the accuracy (8). Logistic regression is a classification model that categorizes datapoints by minimizing the mean square error loss from a linear combination of features. The Support vector machine (SVM) creates a line that maximizes the distance from all the points, but in contrast to logistic regression, SVM's are not necessarily linear. Random Forest is a common machine learning model that combines the output of many decision trees. In the training

set, the most positive features included “thank”, “love”, “yay”, “happy”, and “cant wait”; the most negative features are “sad”, “miss”, “wish”, “sucks”, “hate”. Although these features can be used in a different context than their given sentiment, such as, “I wish he would be happy,”, for most of the time, the sentiment of the rest of the tweet overrides the sentiment of a feature taken out of context. An explanation of the features tested when training these models follows.

Hyperparameters

Table 1 Logistic Regression Hyperparameters and accuracy scores

Ngram Lower Bound	Ngram Upper Bound	Max Features	c	Accuracy Test (%)	Accuracy Train (%)
1	2	5500	0.75	0.7531	0.807675
1	2	6500	0.5	0.7556	0.810075
1	2	6500	0.75	0.7557	0.814125
1	3	6500	0.75	0.7525	0.814665
1	4	6500	0.75	0.7505	0.811387
1	2	6500	1	0.7554	0.816875

The testing of each hyperparameter in the process above is presented in Table 1. Since many features were tested, Table 1 is solely a representation of the hyperparameters, although many more were tested. Similarly, the same hyperparameter process was applied to the SVM and Random Forest methods. In order to test the accuracy scores of hyperparameters, I split the Sentiment140 dataset into a training set which trained the model and a test set that was new to the model. The first three hyperparameters, ngram lower bound, ngram upper bound, and max features were all parameters of the CountVectorizer. Ngram range low and ngram range high specify the possible sequences of consecutive words in text. An upper bound of 2 in the ngram range presented the highest accuracy since a greater number would overfit the model and a lesser number would result in underfitting. Max features is a parameter of CountVectorizer that chooses the features that occur the most frequently and drops every other feature. Finally, the ideal value for max features in this

model is 6500 as shown in Table 1. Finally, the model shown in this table is Logistic Regression, so the C value is an additional hyperparameter which helps with regularization. Regularization adds a penalty for more extreme parameters that can lead to over fitting. The highest accuracy was obtained when C was 0.75.

Handcrafted Features

Next, I added handcrafted features to become a part of the feature matrix. After the first round of testing, I examined individual tweets to find out what would improve the model's accuracy. The two features I used were exclamation percentage and capital percentage. Exclamations can be an important part of classifying tweets as positive or negative instead of neutral. For example, "I want him to win" is less positive than "I want him to win!!!!". However, this was difficult to implement because exclamations could contribute to both the positive and negative sentiment of tweets. I also implemented Capital percentage where "Vote Him" would be 28.6 percent capitol. I added this feature with the same idea as the exclamation percentage, which was not successful for the same reason.

Additional Features

After calculating an initial round of hyperparameters based on accuracy percentages, I tested the model on self-coded tweets taken from the Election Dataset. I performed an extensive error analysis on the first 40 tweets from both Trump and Biden (80 total) and separated them from the rest of the self-coded tweets. For each tweet, I examined its true label, predicted label, cleaned tweet, uncleaned tweet, and the polarities of each feature within the cleaned tweet to find possible features to add. Finally, I tested additional features that could help improve the model.

Table 2: Additional Features from Error Analysis

Test Data	Original	Remove "No", "Nor", and "Not from Stopwords	Fully Cap Words +1	Lemmatize	Removing Numbers	Add Liar
Biden Viewed	0.5125	0.590	0.62	0.513	0.538	0.590
Trump Viewed	0.436	0.487	0.41	0.410	0.436	0.436
Biden Not Viewed	0.4794	0.487	0.49	0.394	0.451	0.437
Trump Not Viewed	0.521	0.507	0.55	0.507	0.507	0.521

Error analysis on viewed tweets as "viewed", and tweets not viewed as "not viewed"

Since “no,” “nor,” and “not” can affect the meaning of a sentence, by removing them from the stopwords list, I did not remove them from the tweet while cleaning the dataset. I hypothesized that this would help classify tweets such as “up to no good” as negative instead of positive, which slightly increased the accuracy. Next, capital words were an essential part of highlighting the most critical parts of a tweet. Therefore, adding an extra instance of every fully capitalized word increased the accuracy percentage of the model. Lemmatization converts words to their root form in the cleaning process; for example, “geese” would be converted to “goose”; however, this process was unsuccessful and decreased the accuracy percentage. Numbers were added as a part of the feature matrix, likely distracting the model without adding any meaning since any specific number could be used in a positive or negative scenario. However, the accuracy percentage did not substantially increase by removing them from the feature matrix. Finally, I added multiple significant words to the feature matrix, including “Liar”, “Corrupt,” “Terrorist,” “Communist,” etc., and each of these words appeared over 20 times in the training data, but adding these words did not significantly change the accuracy scores.

Sentence Embedding

Sentence embedding encodes sentences into vectors that allow for the embedding of full sentences. There are several benefits of using sentence embedding as opposed to a bag-of-words model. Unlike a bag-of-words model, which compares the most common words, Sentence embedding is able to compare the entire sentence that may include rarer words. For example, in “he is a pathetic candidate”, “pathetic” is essential to determine the meaning of the sentence, but it is likely that “pathetic” is not included enough in the training data to be a feature in a bag-of-words model.

I used both tweet-specific cleaned data and basic cleaned data to train the Sentence embedding model. The cleaned data removed mentions following the @ symbol, extra spaces, unknown symbols, links, and other languages besides English and cleaned hashtags the same way as in the bag-of-words method. The basic cleaned data only removed unknown symbols, extra spaces, links, and other languages. Surprisingly, the accuracy obtained with the basic cleaned data was slightly greater than that obtained with the tweet-specific cleaned data.

After cleaning the data, I applied the Universal Sentence Encoder’s sentence embedding models (9). Universal Sentence Encoder encodes text into high dimensional vectors used

for classification and other natural language processing tasks. After using the sentence embedding tool, I trained two machine learning models, LogisticRegression and RandomForest. The Random Forest algorithm resulted in a greater accuracy than Logistic Regression on sentence embeddings since Logistic Regression is a linear model, while the RandomForest is not. The Basic cleaned data applied on a Random Forest Classifier with max features set to 12 presented the highest accuracy scores of 62% on Trump data and 51% on Biden data. However, this accuracy was still not as high as other models tested.

Out-of-box Sentence Classifier

The previous sentiment analysis models do not account for the syntax of tweets. Out-of-box classifiers such as VADER and TextBlob take this syntax into account. For example, the bag-of-words and the sentence embedding models cannot detect sarcasm, a prominent characteristic of tweets, as well as an out-of-box-classifiers can. The ability to fully detect sarcasm would be a useful tool for this study, but it is a narrow research field in NLP and not advanced enough yet. Moreover, tweets are much shorter than traditional text, making it harder for the previous models to predict the sentiment as accurately as the VADER and TextBlob classifiers ostensibly can. VADER is a lexicon and rule-based tool similar to the bag-of-words model but specifically designed for Social Media text (10). A lexicon model converts the text into a bag of words, and each word or phrase is assigned a positive or negative value. Similarly, TextBlob is built from NTLK functions for NLP tasks, including sentiment analysis to provide accurate conclusions regarding the sentiment of sentences (11). I applied both models on cleaned and uncleaned test data and obtained the polarity scores from both. The VADER Cleaned data resulted in an accuracy of 60.9% for Trump and 59.1% for Biden.

Table 3: Comparing the Three Types of Sentiment Analysis Models

Model Type	Accuracy	True Positive	True Negative
Trump Bag of Words	45.5%	59.1%	68.8%
Biden Bag of Words	46.4%	60.5%	60.0%
Trump Embeddings	52.7%	77.2%	52.2%
Biden Embeddings	45.5%	65.6%	65.8%
VADER Trump	60.9%	77.3%	80.4%
VADER Biden	59.1%	83.8%	66.7%
TextBlob Trump	38.2%	62.3%	38.5%
TextBlob Biden	47.3%	86.7%	63.4%

Figure 1 Biden Confusion Matrix

True \ Predicted	-1	0	1
-1	22	8	11
0	5	12	5
1	6	10	31

Figure 2 Trump Confusion Matrix

True \ Predicted	-1	0	1
-1	41	12	10
0	3	9	4
1	5	9	17

I chose to use the VADER classifier based on the accuracy scores and the results from the confusion matrices. The VADER classifier resulted in some of the highest accuracy scores, true positive rates, and true negative rates. Furthermore, using this classifier, the results of both the confusion matrices were significantly similar. Most of the errors of the model were caused by the model overpredicting neutral tweets. Since there is no fine line between what is considered neutral and positive/negative, it was not necessarily a problem that the machine was overpredicting neutral tweets. The classifier also predicted slightly more false negatives than false positives. Although Biden had a higher false negative rate, the same model was applied to Trump and Biden thereby eliminating bias. In the case of tweets, false positives are as deleterious as false negatives; however, when compared with the other models, the VADER classifier was the most accurate. The VADER classifier was the sole

model applied to the election tweets because it provided the best chance of accurately predicting the result of the election.

Related Works

Although tweets present a unique challenge over traditional text because of the shorter length and more informal language, there have been countless examples of sentiment analysis on Twitter tweets using similar methods used in this paper (12-15). Since both writing styles differ significantly, Twitter sentiment analysis techniques are different from longer, more traditional sentiment analysis.

Before introducing sentiment analysis on tweets, classifying text was solely done using raw word representations (ngrams). These were largely successful, as shown in previous papers (16, 17). This paper does include the raw word representations with the bag-of-words method; however as predicted, it did not perform as well in comparison to tweet specific algorithms like the VADER classifier.

Barbosa and Feng created a newer model for creating features specifically for tweets. Instead of traditional n-grams, they use two types of features, meta-features and tweet syntax features (18). Meta-features map features to part-of-speech tags and map words to prior subjectivity. Tweet syntactical features create features based on hashtags, emoticons, upper case words, and other tweet-specific features. In the bag-of-words section, I experimented with some of these features like the number of upper-case words and incorporation of hashtags into the text; however, the out-of-box classifiers like VADER and TextBlob are more similar to the model presented by Barbosa and Feng.

Results

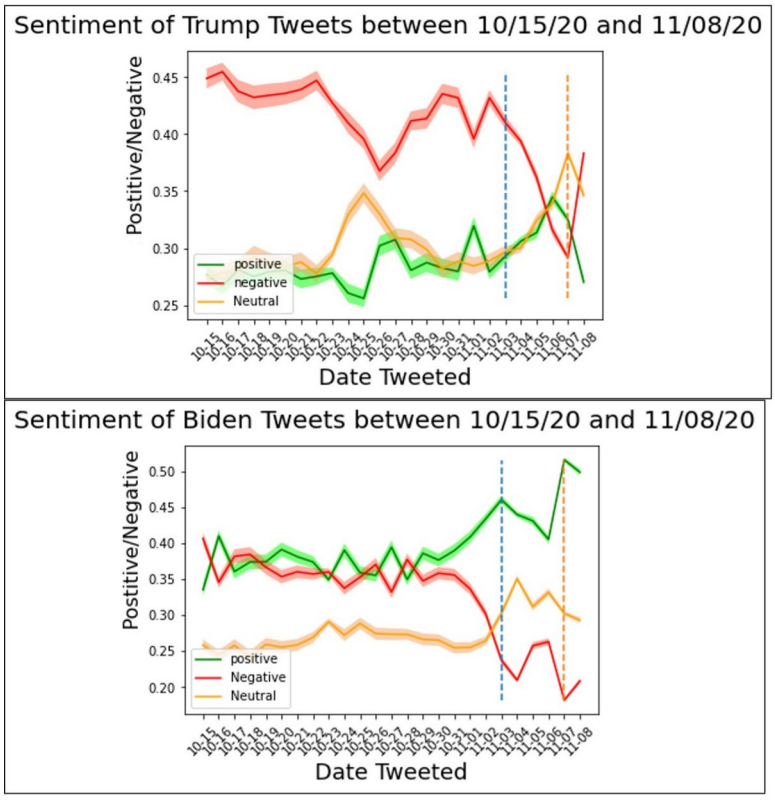
After cleaning the text and removing non-English tweets, I found that there were roughly 541000 Biden-related tweets and 694000 Trump-related tweets. As stated earlier, the dataset of election tweets found Biden tweets using #JoeBiden and #Biden as keywords and found Trump tweets using #DonaldTrump and #Trump as keywords. Therefore, they are a representation of the overall sentiment in terms of both support and opposition, towards the candidate. Out of the Biden tweets, there were about 28% negative, 29% neutral, and 43% positive tweets. On the other hand, out of the Trump tweets, there were about 38% negative, 33% neutral, and 29% positive tweets. Figure 3 show the sentiment percentages per day predicted by the classifier.

Figure 3 can be divided into three sections: before the ballots were cast, after the ballots were cast on 11/3/20, and after the media concluded the election results, on or around 11/6/20 - 11/7/2020. The sections are marked with vertical, dotted lines: the blue line shows

the election day, and the orange line shows when the election was called. As anticipated, at each shift in section, there was a sharp change in the direction of Biden and Trump sentiment, exemplifying the hypothesis that the perception of a candidate changes depending on if they win or lose.

Several papers like (Budiharto and Meiliana, 2014) and (Ceron et al., 2014) have concluded that Twitter sentiment analysis can determine the outcome of the elections in other countries (19, 20). Pre-election sentiment of Trump tweets was more negative than positive, whereas the difference between positive and negative tweets for Biden was significantly smaller. Since the sentiment of tweets is shown to influence the results of an election, the greater ratio of negative to positive tweets for Trump, when compared to Biden, could be a potential predictor for the outcome of the election.

Figure 3 Sentiment of Trump and Biden tweets between 10/15/20 and 11/08/20. Shading on the lines represents the confidence interval.



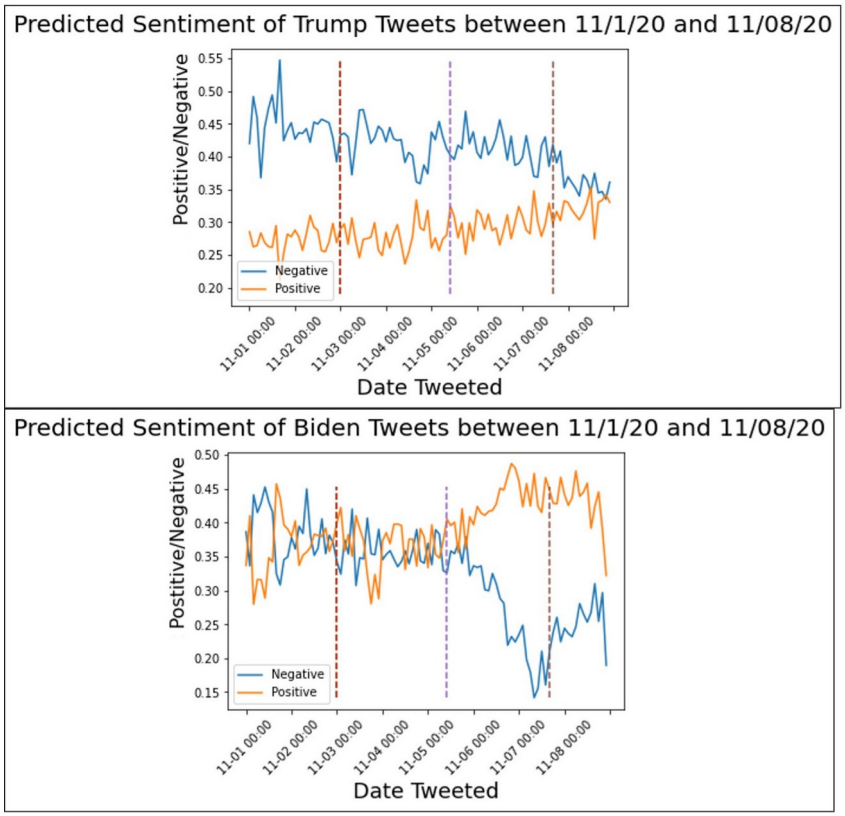
During the period before voting and slightly after on both graphs, there is a noticeable upward trend in positive tweets and a downward trend in negative tweets. After performing qualitative analysis, I found a large portion of tweets during this period (roughly

11/01/20 to 11/03/20 and slightly later in Trump) encouraged voters to vote for their candidate along the lines of "Vote for @DonaldTrump!!" or "Vote for @JoeBiden!!". The increase in positive tweets shows that rather than negatively tweeting about the other candidate, which would have resulted in more negative tweets for both, the election period was a time when people advertised the candidate of their choice, rather than denouncing the opposing candidate.

As expected, after the election results were close to being called (to the right of the vertical orange dotted line), there was a spike in Biden-related positive tweets and Trump-related negative tweets. There is a correlation between the sudden change in trend and the election results, suggesting that the election outcome affected Twitter users since the predicted sentiment changed so drastically.

Figure 3 presented the average predicted sentiment on a day to day basis; Figure 4 shows a more in-depth chart depicting the average sentiment every two hours.

Figure 4 Detailed view of the sentiment of Trump and Biden Tweets between 11/1/20 and 11/08/20



The first red dotted line depicts election day. The second purple dotted line depicts the time Trump tweeted to stop the counting and roughly around when Biden was pulling ahead in the election race. Finally, the third brown dotted line depicts the time of Biden's victory speech.

The Trump sentiment graph shows an interesting trend where initially, the contribution of negative tweets is significantly greater than the predicted positive. As time progresses, they even out when the election is decided. However, after analyzing the tweets of November 8th, it was noticeable that most positive tweets were full of sarcasm. All sentiment analysis models have trouble detecting sarcasm. Almost all of the positive tweets look like "Trump Will Lose the Electoral College & Popular Vote & Still Win (Unraveling Democracy)" or "Well, in all fairness, it is hard for Trumps supporters to accept defeat, because winning was such a sure thing ... for them!". In reality, these comments are sarcastic and negative when keeping Trump's statements of a rigged election in mind. The rest of the tweets classified as negative on November 8th were beliefs that Trump still had the potential to win, and these tweets like "There is strong evidence that in at least 3-4 states the election results have been distorted. Keep fighting Trump!". Therefore, the predicted sentiment of Trump was not adequately represented because of inaccurate classification of sarcastic tweets. However, it could be concluded that a majority of Trump tweets after the conclusion of election day were negative.

The Biden sentiment graph, on the other hand, depicted the opposite relationship to that of Trump's graph. The sentiment of Biden tweets was fairly split between positive and negative for the majority of the election until Trump's statements about stopping the counting, at which point the predicted positive tweets spiked up, and the predicted negative tweets spiked down. After looking at the positive Biden tweets in this period, most were against Trump's statements to stop the counting. For example, many tweets are reassurances that Biden was the victor, such as, "They say a star burns brightest just before it dies, and this was the Trump presidency in all its flaming glory" and "All these questions on how the country can unite...start by accepting the loss. Country Over Party". After the election was called, both the positive and the negative tweets temporarily increased and then started to decrease, probably having to do with how the victory speech was received.

Table 4: CNN Exit Poll

	Favorable	Unfavorable
View of Donald Trump	46%	52%
View of Joe Biden	52%	46%

I also compared the sentiment predicted on election day (the blue line) to the CNN exit polls on both the opinions about Donald Trump and about Joe Biden (21). The exit poll showed that out of the 15,590 randomly chosen respondents, 46% viewed Donald Trump as favorable and 52% viewed him as unfavorable. For Biden, out of the same sample of respondents, 52% view him as favorable and 46% as unfavorable. Favorable and unfavorable in the exit polls correspond to the positive and negative sentiment predicted by the model. A similar trend was shown in the results predicted by the model where Trump had more negative tweets and Biden had more positive tweets; however, the model's classifications showed a slightly more exaggerated difference between the number of positive and negative tweets.

Discussion

Tweets and social media provide a crucial outlook of the election because comments are created based on the individual voter's perspective, not the media, which may be controlled by specific experts and/or interests. It is challenging to accurately predict and/or determine the sentiment of a large population of voters with any other tool than after the actual act of voting. Therefore, the sentiment analysis of Twitter can provide a different temporal perspective of the election than otherwise understood.

Although social media can be an outlook for sharing ideas and giving the ordinary person a voice, it can also be a medium for an increase in extremism where everyone has an opinion that is likely influenced by the majority belief and the candidate they support. Since more polarized tweets receive more attention in the form of comments and retweets, users are encouraged to make more radical statements encouraging extremism. Trump's statement to stop the count had a significant influence on the sentiment of both the Biden and Trump tweets. On the surface, it would have been difficult to imagine the effect his statement had on his followers; however, the sentiment classification visualizes the true power of the candidate's voice. A similar effect can be shown for Biden with the graph of both candidates affected by his victory speech. This herd mentality created by a candidate and fueled by social media forces Twitter users to choose a more radical viewpoint than they otherwise would have, thereby leading to greater political schism. Therefore, the influence of social media can be beneficial in helping a candidate's outreach and influence but may also have detrimental effects such as increasing political polarization. Social

media can also be manipulated by various methods such as paying people or programming bots to write specific Tweets and/or to spread disinformation with a view to influencing the election outcome, such as the Russian Twitter troll farm set up to influence the 2016 US Presidential election. Moderation and/or verification of tweets hence becomes essential for any sentiment analysis algorithm to function as intended.

This model can be applied to elections to gauge the sentiment of a large part of the population (vis-a-vis a small sample) before and after polling which otherwise would be difficult to obtain. It opens new avenues for campaign strategies by reviewing the results of certain events and the impact they have on the general population's beliefs. For example, it would be possible to collect and analyze the public's reaction to a speech or event, either in real time or immediately afterward so as to adjust campaigning strategy 'on the fly'. Furthermore, this paper only used the date on which the tweets were written, to create trends; however, this model can be further applied to obtain the sentiment of entire cities or certain political associations, which would provide immediately actionable and valuable information for campaigns. Similarly, this paper only applied the model to the 2020 US presidential election; however, the same model could be applied to all elections if Twitter data is accessible.

Finally, by analyzing the sentiment of each individual it could be possible to predict future elections using this model. Since the model was able to determine that Trump had more negative sentiment than Biden before the election and Trump ended up having a much lower popular vote, there is a possibility that these events were related. The model can be further tested by applying it to past or future elections using relevant Twitter data for that span of time. The model is currently only able to predict the popular vote, but when further refined, it may be used to predict the results of individual states and the electoral college. Twitter sentiment analysis to predict the outcome of elections could be a useful and powerful tool in the future.

References

[1] Jungherr, A.: Twitter use in election campaigns: A systematic literature review. Journal of Information Technology & Politics 13(1), 72–91 (2016) https://doi.org/10.1080/19331681.2015.1132401. https://doi.org/ 10.1080/19331681.2015.1132401

[2] Bermingham, A., Smeaton, A.: On using twitter to monitor political sentiment and predict election results. In: Proceedings of the Workshop on Sentiment Analysis Where AI Meets Psychology (SAAIP 2011), pp. 2–10 (2011)

[3] Go, A., Bhayani, R., Huang, L.: Twitter sentiment classification using distant supervision. CS224N project report, Stanford 1(12), 2009 (2009)

[4] Hui, M.: US Election 2020 Tweets. https://www.kaggle.com/ manchunhui/us-election-2020-tweets

[5] Vitak, J., Shilton, K., Ashktorab, Z.: Beyond the belmont principles: Ethical challenges, practices, and beliefs in the online data research community. In: Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work amp; Social Computing. CSCW '16, pp. 941–953. Association for Computing Machinery, New York, NY, USA (2016).https://doi.org/10.1145/2818048.2820078. https://doi.org/10.1145/2818048.2820078

[6] Fiesler, C., Proferes, N.: "participant" perceptions of twitter research ethics. Social Media + Society 4(1), 2056305118763366 (2018) https://doi.org/10.1177/2056305118763366. https://doi.org/10.1177/ 2056305118763366

[7] Bird, S., Klein, E., Loper, E.: Natural Langauge Processing with Python: Analyzing Text with the Natural Language Toolkit. "O'Reilly Media, Inc.", (2009)

[8] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B.,Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vander-plas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay,E.: Scikit-learn: Machine learning in Python. Journal of Machine LearningResearch12, 2825–2830 (2011)

[9] Cer, D., Yang, Y., Kong, S.-y., Hua, N., Limtiaco, N., John, R.S., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C., Sung, Y.-H., Strope, B., Kurzweil, R.: Universal Sentence Encoder (2018)

[10] Hutto, C., Gilbert, E.: VADER: A parsimonious rule-based model for sentiment analysis of social media text. Proceedings of the International AAAI Conference on Web and Social Media 8(1), 216–225 (2014)

[11] Ahuja, S., Dubey, G.: Clustering and sentiment analysis on twitter data. In: 2017 2nd International Conference on Telecommunication and Networks (TEL-NET), pp. 1–5 (2017). IEEE

[12] Kumar, S., Srivastava, A.K., Soni, Y., Tyaghi, U., Singh, N.K.: Twitter sentiment analysis: A novel machine learning approach. Int. J. Control Autom. 13(4), 412–421 (2020)

[13] Mittal, A., Goel, A.: Stock prediction using twitter sentiment analysis. Stanford University, CS229(2011) http://cs229.stanford.edu/proj2011/GoelMittalStockMarketPredictionUsingTwitterSentimentAnalysis.pdf) 15 (2012)

[14] Ghiassi, M., Skinner, J., Zimbra, D.: Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural network. Expert Systems with applications40(16), 6266–6282 (2013)

[15] Mishev, K., Gjorgjevikj, A., Stojanov, R., Mishkovski, I., Vodenska,I., Chitkushev, L., Trajanov, D.: Performance evaluation of word andsentence embeddings for finance headlines sentiment analysis. In: Inter-national Conference on ICT Innovations, pp. 161–172 (2019). Springer

[16] Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? sentiment classification using machine learning techniques. arXiv preprint cs/0205070 (2002)

[17] Pang, B., Lee, L.: A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. arXiv preprint cs/0409058 (2004)

[18] Barbosa, L., Feng, J.: Robust sentiment detection on twitter from biased and noisy data. In: Coling 2010: Posters, pp. 36–44 (2010)

[19] Budiharto, W., Meiliana, M.: Prediction and analysis of indonesia presidential election from twitter using sentiment analysis. Journal of Big data 5(1), 1–10 (2018)

[20] Ceron, A., Curini, L., Iacus, S.M., Porro, G.: Every tweet counts? how sentiment analysis of social media can improve our knowledge of citizens' political preferences with an application to italy and france. New media & society 16(2), 340–358 (2014)

[21] National Results 2020 Presidential Exit Polls. Cable News Network. https://edition.cnn.com/election/2020/exit-polls/president/national-results