Sentiment Analysis:Prediction for US Presidential Election 2020 winner using Naïve Bayes and Support Vector Machine Classifier

Aakankshaya Tripathy
Department of Science
Indiana University-Purdue University Indianapolis
aatripat@iu.edu

Raagul Tirupur Senthilkumar
Department of Science
Indiana University-PurdueUniversity Indianapolis
ratiru@iu.edu

Luc Rulinda
Department of Science
Indiana University-Purdue University Indianapolis
lrulinda@iu.edu

Venkatesh Sudireddy
Department of Science
Indiana University-Purdue University Indianapolis
vesudi@iu.edu

Abstract [1]-Sentiment Analysis is a tool used to decipher the opinion of people. The best use of the tool will be in the field of twitter which is directly a form of social talk. The presidential election in the US 2020 is a terrific example where people give their opinion for their favorite candidates on social media and this tool can therefore be used to predict where the verdict sways in terms of their favorite candidate. There are multiple algorithms to predict the score of the outcome and in this paper, we have used multinomial Naïve Bayes and Support Vector Machine classifiers, respectively. The method used for this paper is data collection, data preprocessing, data mapping, and sentiment analysis. The outcome is in the form of a score which will say how the data sets that we have used align with the respective candidates. In this paper, the tweets of the users are given a polarity score depending on the sentiment. Based on the probability of polarity being positive, negative, or neutral for a data set, we train the algorithm to predict a score.

Keywords—Twitter, Kaggle Data Base. Sentimental Analysis, US Presidential Election 2020, Naïve Bayes Classifier, SVM Classifier ***[2] [3]***

INTRODUCTION

· Sentiment analysis (also called opinion mining) refers to the application of natural language processing, computational linguistics, and text analytics to identify and classify subjective opinions in source materials (e.g., a document, a sentence, or a tweet). Generally speaking, sentiment analysis aims to determine the attitude of a writer concerning some topic or the overall contextual polarity of a document. The attitude may be his or her judgment or evaluation, affective state (that is to say, the emotional state of the author when writing), or the intended emotional communication (that is to say, the emotional effect the author wishes to have on the reader). ***[4].*** With the advent of social media, there has been an increase in the way people voice their opinion. A tremendous amount of data representing people's opinions, which is the user content, is available for research. Sentiment analysis is one such method that uses the user content to classify into a positive and negative spectrum of emotions. ***[5]***

· Sentiment analysis is also used by big corporate companies to get feedback from the customers on the product ***[6]***

· Also, it is used as an index to measure the satisfaction of customers on consumer products. ***[7]***

· Sentiment analysis is used in fields of economics, social behavior, politics, and other domains. One important topic is the stock market forecast. ***[8]***

· It is used to detect crime within twitter operations. ***[9]***

· Sentiment analysis can be used in social media data for disaster response management, including enhancing situation awareness, promoting emergency information flow, and predicting disasters, and coordinating rescue efforts. [10]

· In this paper, we are using sentiment analysis in election prediction, a widely studied topic. We use Twitter and tweets as input to understand users 'political preferences and inclinations.

· Using the tool, we can understand which tweets are positive, negative, or neutral to the candidate. Predicting this can give an idea of the outcome of the elections.

· In our work, we focused on Twitter sentiment analysis of the 2020 U.S. Presidential Election between Donald Trump and Joe Biden. We used two-sample data sets collected from Kaggle to train the model and we used Naïve Bayes and Support Vector Machine as the classifier with the highest performance. We then fitted the model with tweet data on the 2020 U.S. Presidential Election collected. . The score for Joe Biden's win comes out be 86.96% while the score for Donald Trump comes out be 81.5% using multinomial Naïve Bayes Classifier. And 85 % for Biden and 84% for Trump using SVM classifier. ***[1]***

METHODOLOGY

In this paper, we first introduce the data sets for sentiment analysis. Then we use two different machine learning applications called Naïve Bayes Classifier and Support Vector Machines. At the end of the paper, we have compared the results from the two classifiers to understand the best algorithm for this approach.

PRELIMINARY BACKGROUND

· A Naive Bayes classifier is a probabilistic machine learning model that is used for classification tasks. The crux of the classifier is based on the Bayes theorem. Using Bayes theorem, we can find the probability of A happening, given that B has occurred. Here, B is the evidence and A is the hypothesis. The assumption made here is that the predictors/features are independent. That is the presence of one feature does not affect the other. Hence it is called naive. [11]

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

· A support vector machine (SVM) is a machine learning algorithm that analyzes data for classification and regression analysis. SVM is a supervised learning method that looks at data and sorts it into one of two categories. An SVM outputs a map of the sorted data with the margins between the two as far apart as possible. SVMs are used in text categorization, image classification, handwriting recognition, and in the sciences [12]

DATA COLLECTION

· In this approach, we first introduced two CSV data sets from Kaggle. One data set represents all the tweets collected for Joe Biden and the other data set represents all the tweets collected for Donald Trump. Tweets were classified as positive, negative, and neutral based on sentiments. We used 5000 tweets for the analysis from the data sets. [2]

PREPROCESSING

· This section deals with cleaning the tweets from the datasets before doing sentiment analysis. The following steps are followed for the same:

- Changing the case of all the tweets to lower case
- Replacing the unwanted characters and strings including URL from tweets.
- Tokenization
- Keyword Search for each data set.

SENTIMENT ANALYSIS

- Using the TextBlob library, we import the sentiment function to get the sentiment of each cleaned tweet. Each tweet can give 3 scores 0 – negative, 1- neutral, and 2 – positive.
- We will have two sets of data – one with tweets and the other with the polarity of each tweet.
- Based on the sklearn library function, we split the data and create train and test sets.
- Further we vectorize the data set.
- Finally, we pass the classifiers to give us a score on how the probability of the features performs across each tweet label.
- The better this happens more the efficiency of the classifier.

EXPERIMENTAL SETUP

For this paper, we have used the data set to run a sentimental analysis on it. It is done after cleaning the tweets and extracting the features. We create a train and test data set from label and feature data. Features are based on the polarity of each tweet based on the meaning of words in the tweet.

STEP 1 –

Collected User Data –

- Static data set from Kaggle which has 500000 tweets from Biden and Trump cases. In this model, we have used only 5000 from each for analysis. [2]
- Using Tweepy (Twitter API) we can find tweets using keywords from a particular date range and use the classifier on that. [4]
- All the tweets from both data sets were preprocessed before taking the 5000 tweets into account.

Table 1: Sentiment Analysis Using Text Blob Sentiment [5]

Sentiment	Tweet for Joe Biden
Positive (1)	"I'm going to share things I like about Biden more. you should too. Biden cares Biden chicken trump Kamala Harris"
Negative (-1)	"FBI allegedly obtained hunter Biden computers, data on ukraine dealings, report claims joebiden hunterbiden"
Neutral (0)	"elecciones2020 | en florida: joebiden dice que donaldtrump solo se preocupa por él mismo. el demócrata fue anfitrión de encuentros de electores en pembrokepines y miramar. clic aquí ↓↓↓ elsollatino yobrilloconelsol

STEP 2 –
PREPROCESSED THE DATA

To make our data sets usable, we need to clean the tweets for input. This is because there are several strings and URLs which do not do justice in helping us determine the polarity.

Replacing the unwanted strings [13]

Multiple special characters are used in tweets which is not important while sentiment analysis, for example, a URL or Hashtag will not tell us anything about the sentiment of a tweet.

Table 2: Predefined Strings

Underscore	'_'
URL	"https:\S+"
Line Change	"\n"
HASH	"#"
Address to	"@"
EXCLAIM	"!"
Numbers	"0-9"

Tokenization [5]

Using wordnet, identifying parts of speech in tweets, and counting those words helps make the tweets clean. `'J': wordnet. ADJ, 'V': wordnet. VERB, 'N': wordnet. NOUN, 'R': wordnet.ADV}`

Keyword Search [13]

For Data Sets, the Keywords Used Are

Biden KEYWORDS	TRUMP KEYWORDS
'Biden' ,' joe' ,' blue' ,' Kamala ,' corona' ,' democrats'	'trump', 'Donald' 'red' 'maga' 'republican' 'fakenews' 'communism'

Table 3: Search Keywords

STEP 3 – CALCULATION OF POLARITY

Then we used Text Blob sentiment analysis to find the polarity score of every tweet to break down the sentiment into Positive, Negative, and Neutral. [14] [13]

Polarity	Biden Count	Trump Count
Neg	1048(20.9%)	1207(24%)
Pos	1522(30.45%)	1593(31.8%)
Neu	2427(48.55%)	2199(43.9%)
Total	4997(99.9%)	4999(99.9%)

Table 4 - Tweet Sentiment Analysis on Biden and Trump using sentiment analysis

STEP 4 – CREATION OF TRAIN AND TEST DATA ·

We then split the data set into X and Y train and test data sets using the train_test_split module from sklearn. This creates a test data set from the training data set in the ratio of 1:3. [14]

STEP 5- VECTORIZE TEXT DATA

- Using sklearn import the CountVectorizer function from sklearn to vectorize the tweets into 0 (negative), 1(neutral), 2 (positive) arrays based on the polarity score calculated above. [14]

STEP 6 – USING NAÏVE BAISE CLASSIFIER OR SUPPORT VECTOR MACHINE CLASSIFIER TO PREDICT

- Using the multinomial sklearn library we import the NB classifier and use it to fit and predict the score for the train and test data sets.
- In the case of SVM, we invoke the linear SVC function to predict the outcome.
- The final process is used to use the Naïve Bayes classifier and Linear SVM on the test vs train data set to predict the probability of the win percentage. [14]

RESULT ANALYSIS USING NAÏVE BAYES CLASSIFIER

Table 4 shows the analysis done by using Sentiment Polarity. We applied the classifier on 5000 data points from each Biden and Trump candidate. In the Biden data set, nearly 1048(20%) of the tweets were negative and 1522(30.45%) of the tweets were pro-Biden.

On the other hand, in the Trump data set, 1207(24%) of the tweets were negative and 1593(31.8%) of the tweets were positive.

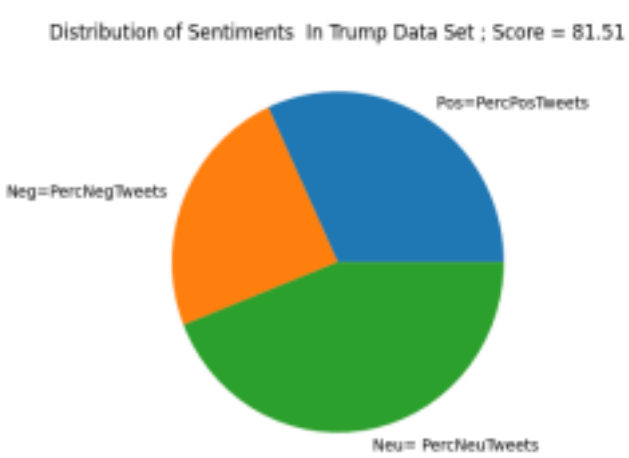
Both had 2427 and 2199 tweets which were neutral. This may be due to high nonnative opinions from outside of the USA which has cast their opinions, not in the English Language.

Both had a fair share of negative tweets with Trump slightly higher. This could be due to people just venting their opinion on the existing policies or Twitter acting as a medium to highlight negative opinions more than positive. This is a common pattern among various social media platforms.

Both had a close share of positive tweets with Trump slightly high. This may be due to the reason that Trump supporters use social media as a major tool to voice their opinions. Also, the fact that Donald Trump was the existing president, he had more supporters online as compared to Biden.

TRUMP DATA SET PLOTS

BIDEN DATA SET PLOTS



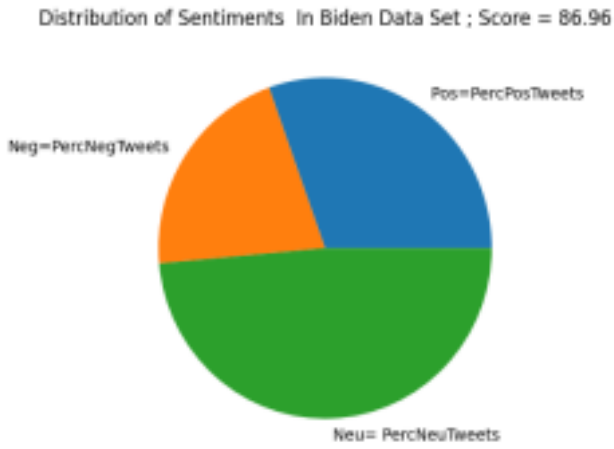
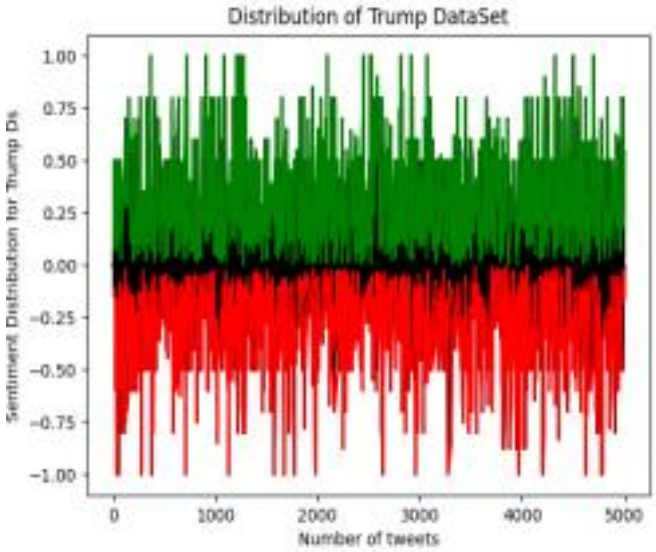


Fig1: Tweet Distribution based on sentiments with NB
Fig 3 – Tweet Distribution based on sentiments using

Fig2: Tweet Distribution based on sentiments with NB



Polarity per tweet

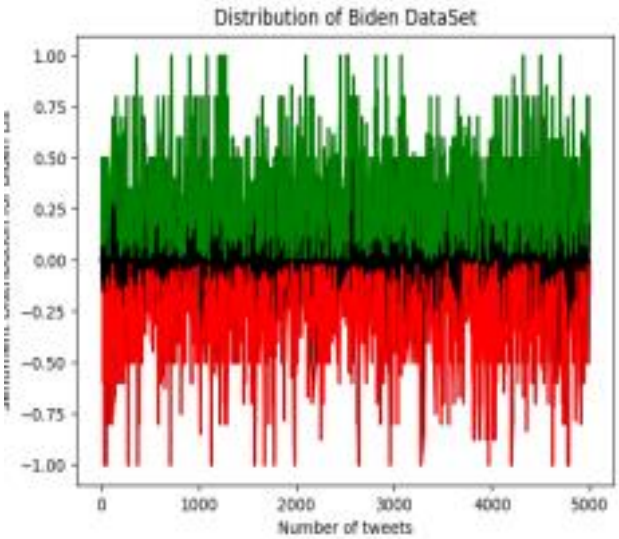


Fig4: Polarity per tweet

RESULT ANALYSIS USING SVM CLASSIFIER

The SVM classifier was also trained using 67% of the entire five thousand Data points and evaluated with the rest 33%. The Biden dataset has a slightly better accuracy score of 84% compared to Trump's dataset with 83.5%. Overall results of our classifier reflect the initial dataset's sensitivity percentages. The Trump dataset has a biggest negative percentage of the two datasets, and the same thing still applies in SVM's classification results: 16.4% in the Trump dataset compared to 10.64% in the Biden dataset (figure 5 and 6). Trump's dataset also still has a better positive category percentage compared to Biden's.

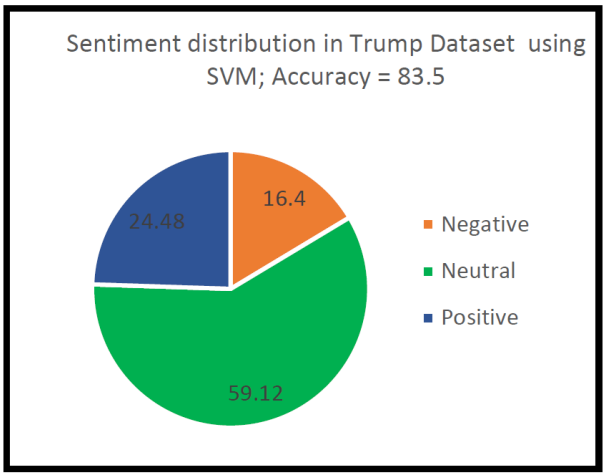


Fig5: Tweet Distribution based on sentiments with SVM

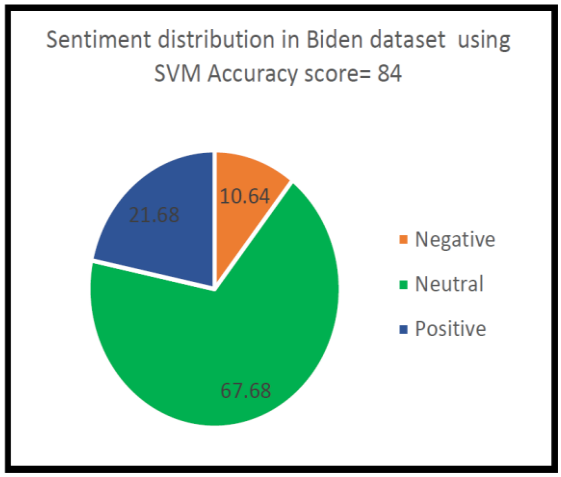


Fig6: Tweet Distribution based on sentiments with SVM

.

CLASSIFICATION PERFORMANCE

In this paper, we used two methods to see which gives better performance in prediction. We used Naïve Bayes Classifier and Support Vector Machines to predict the score for each data set. [15] [16]

Classifier	Efficiency	Biden Data Set	Trump Data Set
Naïve Bayes	**85%**	**86.97%**	**81.5%**
SVM	**84.5%**	**85%**	**84%**

Table 5 - Classifier Performance

Caveat:

· Some tweets can be given different polarities based on how the tweet is written.

· The biasing accounts for 2-3% of the tweets in a set of 5000.

· The number of neutral tweets is high in the model is because of non-English words in the tweets. [14]

COMPARISON WITH COMPETING METHODS [1]

After a comparison of our results with the competing paper referenced, we found that our numbers in terms of positive and negative sentiment tweets are close to the power. We have used five thousand data points and because of the proficient level of preprocessing, we can build a more efficient and accurate model using the NB classifier. The accuracy score obtained from the SVM classifier (83.5% and 84%) in our paper is much better compared to the 54.3% found in the competing paper 's experiment. Another observation that should be noted is that the best accuracy scores in both papers belong to the NB classifier. The best score obtained in our experiment (86.96) and the one in the competing paper (77.13%) both came from the NB classifier. Therefore, when it comes to accuracy both papers' results show that the Naive Bayes classifier is the better classification algorithm.

- **Runtime for Naïve Bayes Classifier (5000 data point) ~= 55s**
- **Runtime for SVM Classifier (5000 data point) = 40s**

Classifier	Sentiment	Efficiency	Efficiency Our Paper	Biden	Our Paper	Trump	Our Paper
Naïve Bayes	**Positive**	**77.13%**	**85%**	**33.35%**	***30.45%***	**34.27%**	***31.80%***
	Negative	**77.13%**	**85%**	**66.65%**	***68.12%***	**65.73%**	***69.54%***
SVM	**Positive**	**54.34%**	**85%**	**31.1%**	**21.68%**	**31.9%**	**24.48%**
	Negative	**52.34**	**85%**	**68.9%**	**78.32%**	**68.1%**	**75.52%**

Table 6 - comparison with competing methods

DISCUSSION and FUTURE SCOPE

In this paper, we used a sentimental analysis tool to predict US 2020 presidential outcomes. We have used two classifiers Naïve Bayes Classifier and Support Vector Machine classifier to find which predicts the best. From the classifier performance in table 5,
It shows Naïve Bayes classifier performs better as compared to the SVM classifier in terms of classifier efficiency and outcome.

According to our calculations, the outcomes were very close to each other on social media using both classifiers, and it was difficult to predict a winner till the results are out. This was proved in the outcome of the presidential election which was a close race.

Using the tool in the future, we recommend using more filtration to extract higher features especially from neutral data sets which will give us a more robust calculation outcome.

CONCLUSION

This paper aims to predict the winner of US Presidential efficiency. However, the efficiency score was 84.5% using the Support Vector Machine algorithm. Using the Naïve Bayes classifier, the Biden dataset score was 86.97%, and the Trump dataset score was 81.5%. Using the Support Vector Machine classifier, the Biden dataset score was 85%, and the Trump dataset score was 84%. Also, the Naïve Bayes classifier's execution time is 55 seconds, and the Support Vector Machine classifier's execution time is 40 seconds. The difference in execution time is due to extra pre-processing steps in the Naïve Bayes classifier that resulted in better output than the Support Vector Machine classifier. These comparisons are clear that the Naïve Bayes algorithm displayed an almost accurate margin win of the actual US Presidential Election 2020 results. Election 2020 results using tweets from Twitter using Naive Bayes and Support Vector Machine(SVM) algorithms in Python. The output of our sentiment analysis resulted in terms of overall efficiency represented in %. We compared the Naïve Bayes and Support Vector Machine algorithms for sentiment analysis of two unique datasets containing 60k tweets of Joe Biden and Donald Trump from Kaggle. For preprocessing, we used five thousand tweets about elections. Preprocessing consisted of cleaning the data, filtering stop words, and matching related keywords. It played us move a step ahead towards our goal. The cleaned datasets were split into a 1:3 ratio of Test and Train datasets. Polarity played a vital role as it helped us identify the sentiment of keywords by categorizing words from the tweets into positive(1), negative(0), and neutral(2). The results of these classifiers depicted better results with the Naïve Bayes algorithm with 85%.

References

[1] "Xia, Ethan, Han Yue, and Hongfu Liu. "Tweet Sentiment Analysis of the 2020 US Presidential Election." Companion Proceedings of the Web Conference 2021. 2021.".

[2] "https://www.kaggle.com/manchunhui/us-election-2020-tweets," [Online].

[3] "Oikonomou, Lazaros, and Christos Tjortjis. "A method for predicting the winner of the usa presidential elections using data extracted from twitter." 2018 South-Eastern European Design Automation, Computer Engineering, Computer Networks and Society Media C".

[4] "https://www.researchgate.net/publication/300495226_Sentiment_Analysis".

[5] "https://medium.datadriveninvestor.com/predicting-us-presidential-election-using-twitter-sentiment-analysis-with-python-8affe9e9b8f," [Online].

[6] "https://www.intellectyx.com/sentiment-analysis/," [Online].

[7] ""https://www.researchgate.net/publication/323536432_Customer_Satisfaction_Measurement_using_Sentiment_Analysis".".

[8] ""https://towardsdatascience.com/sentiment-analysis-for-stock-price-prediction-in-python-bed40c65d178".," [Online].

[9] "chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/viewer.html?pdfurl=http%3A%2F%2Ferepository.uonbi.ac.ke%2Fbitstream%2Fhandle%2F11295%2F76651%2FDende_Sentimental%2520Analysis%2520in%2520Crime%2520Detection-%2520a%2520Case%2520Study%2520of%2520Kenya%252," [Online].

[10] ""https://www.sciencedirect.com/science/article/pii/S2212420921000674".," [Online].

[11] "https://towardsdatascience.com/naive-bayes-classifier-81d512f50a7c".," [Online].

[12] "https://www.techopedia.com/definition/30364/support-vector-machine-svm".," [Online].

[13] ""https://towardsdatascience.com/sentiment-analysis-on-twitter-data-regarding-2020-us-elections-1de4bedbe866".," [Online].

[14] ""https://www.analyticsvidhya.com/blog/2021/07/performing-sentiment-analysis-with-naive-bayes-classifier/"," [Online].

[15] ""https://www.kaggle.com/langkilde/linear-svm-classification-of-sentiment-in-tweets".," [Online].

[16] ""https://www.pluralsight.com/guides/building-a-twitter-sentiment-analysis-in-python".," [Online].