

Final Project Report – US Election 2020 Sentiment Analysis

Introduction:

Forecasting the Presidential Elections has become a trend in Academia. Recently, US 2020 Elections were held, and Joe Biden was declared the winner. I wanted to know if this can be predicted from the twitter data before the election results were declared. The main contestants were Joe Biden from Democratic party and Donald Trump from Republican Party. I wanted to do Sentiment Analysis on Twitter data to get results of who will be the possible winner of US 2020 Elections.

Problem Statement:

- Is Trump liked more or Biden?
- Is Trump or Biden, who is more popular in India?
- How are the sentiments for each leader among the people?

Related Work:

In last decade, there was a steady increase in data mining and finding out the opinions in text data very much. This particular opinion data from Twitter is used for stock market prediction and also for analyzing Social movements across United states and many other countries. Tumasjan (2010) found tweet volume about the political parties to be a good predictor for the outcome of the 2009 German election, while Choy et al. (2011) failed to predict with Twitter sentiment the ranking of the four candidates in Singapore's 2011 presidential election. Now, I am using data from Kaggle, which is very apt for this project and would like to get good results from this data for deep understanding of the cultural and political practices at work through the use of Twitter data.

Methods:

Tools used -

I worked with Python Programming Language in IDE Google Colab. I used Pandas for Reading the Data set and also for Data Transformation, NumPy also for data Transformation. Matplotlib for Data Visualizations.

Load Data –

First, I loaded the Two data sets “Trump Data”, “Biden Data” into two different data frames.

```
df_Trump = pd.read_csv('/content/hashtag_donaldtrump.csv',lineterminator='\n')  
df_Biden = pd.read_csv('/content/hashtag_joebiden.csv',lineterminator='\n')
```

Data Transformation –

I drop () function to drop few columns in the data set as they were not useful for my analysis.

I used loc [] method to get only the data of Trump in Trump Dataset and Biden in Biden Data set.

I used Concat () method to mix both the data frames into one data frame.

I used Sort_values () method to sort the data frame on the column “created_at” in ascending order.

```
: #Remove the columns not used  
df_Trump = df_Trump.drop(columns = ['tweet_id', 'collected_at', 'user_description', 'collected_at'])  
df_Biden = df_Biden.drop(columns = ['tweet_id', 'collected_at', 'user_description', 'collected_at'])
```

```
: df_Trump.loc[:, 'conidat'] = 'Trump'  
df_Biden.loc[:, 'conidat'] = 'Biden'
```

```
: df_Mix = pd.concat([df_Trump, df_Biden])  
df_Mix.sort_values(by='created_at')  
df_Mix.head()
```

Data Visualization –

Xtick () function is used in bar plot to keep labels in x axis. Here I used Xticks for labeling Donald Trump and Biden on x-axis.

Used Bar plot from Matplotlib for this.

Here, I used default blue color as we don't have many variables to compare to.

```
Trump_tweets = df_Mix.query('(conidat == "Trump")').likes.count()  
Biden_tweets = df_Mix.query('(conidat == "Biden")').likes.count()  
  
objects = ('Donald Trump', 'Joe Biden')  
y_pos = np.arange(len(objects))  
performance = [Trump_tweets, Biden_tweets]  
  
plt.bar(y_pos, performance, align='center', alpha = 0.5)  
plt.xticks(y_pos, objects)  
plt.ylabel('Number of Likes')  
plt.title('Donald vs Biden')  
  
plt.show()
```

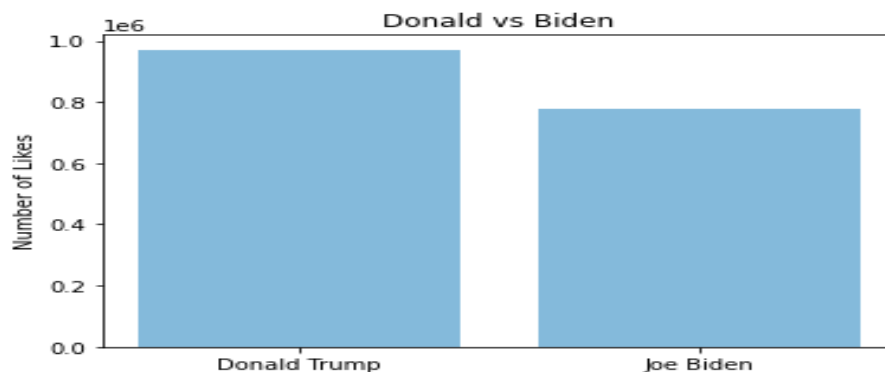


Figure: Number of Likes between Donald Trump and Joe Biden

Tweets count of each candidate is visualized through pie chart. Here I defined a function to calculate the pie chart with input of country name.

```
def Country_tweets(country,kind):  
    Data=df_Mix[df_Mix.country == country].groupby('candidate').tweet.count()  
    Data.plot(x = 'candidate',y = "tweet",kind = kind, title="tweets count in country " + str(country))  
  
Country_tweets('India','pie')
```

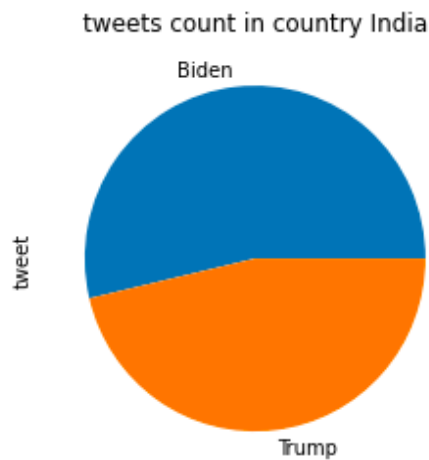


Figure: Number of Tweets count in India of Donald Trump and Joe Biden

Orange and Blue colors differentiate each candidate in pie chart.

Data Cleaning –

In order to do Sentiment Analysis, we have to remove the noise in the tweets like punctuation, numbers, stop words, spaces and convert them to lower case.

```
def clean(text):  
    '''Make text lowercase, remove text in square brackets,remove links,remove punctuation  
    and remove words containing numbers.'''  
    text = str(text).lower()  
    text = re.sub('[.*?\\]', '', text)  
    text = re.sub('https?://\\S+|www\\.\\S+', '', text)  
    text = re.sub('<.*?>', '', text)  
    text = re.sub('\\n', '', text)  
    text = re.sub('\\w*\\d\\w*', '', text)  
    text=re.sub(r'@[A-Za-z0-9]+', '',text)  
    text=re.sub(r'#', '',text)  
    text=re.sub(r'RT[\\s]+', '',text)  
    text=re.sub(r'[^\\w]', ' ', text)  
    return text
```

Dropped the Null Values using function dropna ()

Sentiment Analysis:

I have created functions to calculate the subjectivity score, polarity score and analysis based on score to find the sentiment of the tweets.

Here I had used NLTK, TextBlob Libraries for finding the scores.

```
# create fuction to get the subjectivity and polarity
def getSubjectivity(text):
    return TextBlob(text).sentiment.subjectivity

def getPolarity(text):
    return TextBlob(text).sentiment.polarity

def getAnalysis(score):
    if score < 0:
        return 'negative'
    elif score==0:
        return 'neutral'
    else:
        return 'positive'
```

Imported Counter from Collections library to count the values of each sentiment positive, negative and neutral.

```
from collections import Counter

target_cnt = Counter(Trump_Tweets.polarity)

plt.figure(figsize=(16,8))
plt.bar(target_cnt.keys(), target_cnt.values())
plt.title("Dataset labels distribution")
```

I imported graph_objects from plotly library to make a funnel area chart.

```
import plotly.graph_objects as go
general = Trump_Tweets.groupby('analysis').analysis.count()
neutral = Trump_Tweets[Trump_Tweets['analysis'] == 'neutral'].ClearTweet.count()
positive = Trump_Tweets[Trump_Tweets['analysis'] == 'positive'].ClearTweet.count()
negative = Trump_Tweets[Trump_Tweets['analysis'] == 'negative'].ClearTweet.count()

fig = go.Figure(data = [go.Funnelarea(labels = ["positivity", "negativity", "neutrality"], values = [positive,negative,neutral]))])
fig.update_layout(title_text = 'sentimat analysis tweets Donald Trump')
# fig.show()
display(fig)
```

Funnel Area is used to show data in different stages.

sentimat analysis tweets Donald Trump

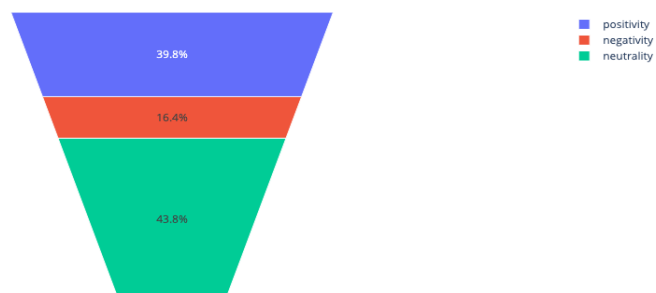


Figure: Funnel Area Graph to represent sentiments in tweets of Donald Trump

sentimat analysis tweets Joe Biden



Figure: Funnel Area Graph to represent sentiments in tweets of Joe Biden

We can clearly see from the chart that Trump has 39.8%, Joe Biden has 45.5% positive tweets.

Results:

From the sentiment analysis, we found out that Joe Biden has more chance of winning the US 2020 Election even though the data had more likes for Donald Trump.

Discussion:

From the Visualizations people can infer that Donald Trump has more likes than his counterpart from the bar chart. The Pie chart tells that In India, there was more discussion about Joe Biden than Donald Trump. From Sentiment Analysis, through funnel area graph chart we found that Donald Trump has fewer positive sentiments and more negative sentiments in the tweets than Joe Biden.

Future Work:

We should have ideally same number of tweets from all the states, well that is quite impossible to have that, but we can try by Distributing the data Uniformly. We can also use Supervised and Un Supervised Machine Learning Algorithms predict the sentiments with better accuracies.

References:

Wang, Hao, et al. *A System for Real-Time Twitter Sentiment Analysis of 2012 U.S. Presidential Election Cycle*.

Gukanesh, A. V., Kumar, G. K., & Saranya, K. K. R. K. | N. (2018). Twitter Data Analytics – Sentiment Analysis of An Election. *International Journal of Trend in Scientific Research and Development*, Volume-2(Issue-3), 1600–1603.
<https://doi.org/10.31142/ijtsrd11457>

KA, A. (2020, November 12). Data Visualization: US Election 2020. Retrieved from Medium website:

<https://medium.com/swlh/data-visualization-us-election-2020-a3d3531b611c>