TUNKU ABDUL RAHMAN UNIVERSITY COLLEGE

BEYOND EDUCATION

BACS3074 ARTIFICIAL INTELLIGENCE

202006 Session, Year 2020/21

Assignment Documentation

Full Name: WONG YEW LEE

Student ID: 19WMR06837

Programme: RDS2

Tutorial Class: GROUP 1

Project Title: ***SENTIMENT ANALYSIS ON TWEETS RELATED TO PRESIDENT TRUMP***

Module In-Charged: ***SENTIMENT ANALYSIS USING VALENCE AWARE DICTIONARY AND SENTIMENT REASONER(VADER) LIBRARY***

Other team members' data

No	Student Name	Module In Charge
1	LOH JIA CHENG	TWITTER TWEETS CRAWLER
2	LI CHEN ZHEN	SENTIMENT ANALYSIS USING TEXTBLOB LIBRARY

Lecturer: Dr. Lim Yee Mei	**Tutor : Dr. Lim Yee Mei**

Deadline: 11th September 2020 (Week 13, Friday, turn in to Google Classroom before 11.59pm)

1. Introduction

1.1. Problem Background

Donald John Trump, the 45th president of the United States is one of the controversial world leaders as of now. Before entering politics, he was a businessman and television personality. Better known as Donald Trump or President Trump, was elected as 45th President of the United States even though he did not win the popular votes of the people. Trump often makes controversial moves throughout his political career and he is very active on Twitter. Trump has often made it to the trending top list of Twitter.

Donald Trump frequently tweets extreme tweets which draw many comments from all of the both spectrum of politicians. Donald Trump has often received highly polarized comments which are very suitable to do analysis. Therefore, we are planning a sentiment analysis on tweets related to the keyword and hashtag related to ”Trump” OR “realDonaldTrump” OR “Donald Trump”.

Twitter is a perfect data source for public sentiment analysis as the tweets are shown for everyone, with a maximum of 140 characters. Just enough for someone to convey a single idea or opinion about a particular subject in a simpler form.

Sentiment Analysis is a study of positiveness or negativeness on a certain sentence mostly for feedback, rating and comment of a certain product or service. Since politicians may receive a heavy load of feedback especially when their statement is released, manpower cannot process all of it in a very short time. Sentiment analysis can be defined as the process of computationally identifying and categorizing opinions from a piece of text and to determine whether the writer’s attitude towards a particular topic or the product, is positive, negative or neutral (edureka.co ,2018)

1.2. Objectives/Aims

- To extract and gain valuable insights from raw data, while gaining an overview of general public opinion on Donald Trump.
- To analyze the sentiments of posts related to Donald Trump.
- To forecast the possibility of him winning the presidential election in 2020.
- To evaluate the performances and satisfaction of Americans towards their president.

1.3. Motivation

Donald Trump is seeking Presidential Reelection in 2020. By conducting a sentiment analysis on social media public opinion, we can forecast the possibility of Donald Trump winning the next election which is in 2 months time. Other than this, by analyzing public perception, the outcome of this analysis can be sold to Donald Trump Election Campaign for them to plan their campaign strategy. Campaign teams can better understand the public opinion on a bigger scale through data visualization, which is very difficult to achieve without the help of computers. Election teams can also measure the ROI of their election campaigns and improve their reach.

The sentiment analysis report can also be used by Donald Trump's Public Relation Team for damage control especially against those negative tweets. Negative tweets can be pulled and analyzed to better understand and improve their campaign.

On the perspective of social impact, Donald Trump knows the feedback from the policy he made in his office through sentiment analysis on a certain time period. This can lead to a better policy formulation for the people of the United States. With the ability to monitor and analyze the conversations happening on social media and beyond, he can use those feelings to make actionable decisions on behalf of the administration.

1.4. Timeline/Milestone

Week	Progress
1	Selecting project title and research about sentiment analysis
2	Applied for Twitter developer account under academic/student purpose
3	Familiarized with the Twitter developer dashboard and read Twitter developer documentation
4	Experimenting with different ways of data extraction from Twitter
5	Exploring tweets scraping using Selenium and BeautifulSoup
6	Able to scrape tweets from Twitter using Twitter API, exploring ways on how not to scrape truncated tweet text
7	Exploring and learning ways to do sentiment analysis
8	Exploring different libraries of sentiment analysis and reading

	conference papers on VADER
9	Experimenting by performing sentiment analysis on datasets obtained from Kaggle
10	Perform sentiment analysis on datasets crawled by TwitterAPI
11	Visualize outcome and fine tuning the code
12	Documentation

2. Research Background

2.1. Background of the applications

Sentiment analysis is useful in a wide range of problems, particularly that are of interest to human-computer interaction ranging from sociology to political. A comprehensive and good lexicon is important for fast and accurate sentiment analysis on such large scales.

To perform a sentiment analysis, a predictive model must first be made. Machine learning is implemented to create this model and solve this kind of issue by using what it is called a polarity of the sentence where it can be easily recognized as marks given by the machine based on words that represent good rating and a bad rating such as good, great, awesome, wonderful, terrible, awful, useless and so on. (edureka.co ,2018)

Sentiment analysis can be defined as the process of computationally identifying and categorizing opinions from a piece of text and to determine whether the writer's attitude towards a particular topic or the product, is positive, negative or neutral. (edureka.co,2018) Sentiment Analysis consists of 5 steps which can be used to determine whether the feedback is good or bad, it is tokenization, cleaning the data, removing stop words, applying supervised algorithms for classifications and calculations. (edureka.co,2018)

The pre-built model i am using in this sentiment analysis is Valence Aware Dictionary for Sentiment Reasoning, better known as VADER is a parsimonious rule-based model for sentiment analysis of social media text. This sentiment analysis model is specifically tuned to social media content that is expressing sentiment. By comparing this rule-based model for sentiment analysis, its effectiveness outperformed individual human raters by 0.12 in F1 Classification Accuracy. It is generalized more favorably across context than any of the existing benchmarks. (Hutton & Gilbert, 2014)

According to a paper published in Eighth International Conference on Weblogs and Social Media (ICWSM-14) in 2014, VADER were developed using a combination of qualitative and quantitative methods to product and validate empirically a great sentiment lexicon attuned to microblog_like contexts. It was found that by incorporating heuristics such as grammatical and syntactical conventions, it can improve the accuracy of sentiment analysis engine across many domain contexts. VADER retains and improves on the benefits of traditional sentiment lexicons

such as LIWC. Just like LIWC, VADER sentiment lexicon has been validated by humans. VADER can be considered more sensitive to sentiment expression in the context of social media. VADER was made freely available to all.

Firstly, a human-centred approach was used to construct and validate a valence-aware sentiment lexicon. Screening, training, selecting and data quality checking was done on the crowd-sourced evaluations and validations. Generalizable heuristics humans identification was used to assess sentiment intensity in text and controlled experiments were done to evaluate the impact of grammatical and syntactical heuristics. Finally, ground truths were used in multiple domain contexts to validate the lexicon and gold standards (human-validated) were obtained.

To summarize, VADER performed as well as(an in most cases, better than) eleven other highly regarded sentiment analysis tools. The results highlighted the gains to be made in computer science when the human is incorporated as a central part of the development process. (Hutto & Giber, 2014)

2.2. Analysis of selected tool with any other relevant tools

Tools comparison	**Remark**	**VADER Sentiment Analysis (Python Library)**	**Textblob (Python Library)**	**Rapid Miner**
Type of license and open source license	State all types of license	Free and open-source (MIT License)	Free and open-source (MIT License)	Proprietary
Year founded	When is this tool being introduced?	June 2014	Aug 13, 2009	2006
Founding company	Owner	C.J Hutto & Eric Gilbert	Multiple open-source developers	Rapidminer Inc.
License Pricing	Compare the prices if the license is used for development and business/commercialization	Free for all to use	Free for all to use	License needed to be bought
Supported features	What features does it offer?	● Works excellent with social media context ● Does not require model-training ● Fast and efficient ● Provides accurate threshold for neutral sentiment ● Return sentiment in compounded scores	● Noun-phrase extraction ● Part-of-speech-tagging ● Sentiment analysis ● Tokenization ● Parsing ● N-grams ● word-inflection	● Full transparency and governance for machine learning ● End-to-end collaboration platform ● platform build visual and analytics

Common applications	In what areas this tool is usually used?	VADER were mostly used to perform sentiment analysis on Social Media context.	Textblob were used mostly on official text reviews, movie and product reviews, and also social media text.	Rapidminer was used by many data scientists to perform machine learning, data mining and etc.
Customer support	How the customer support is given, e.g. proprietary, online community, etc.	Github and the open-source community.	Github and the open-source community.	Online customer support by RAPIDMINER Inc.
Limitations	The drawbacks of the software	Does not show the subjectivity of a sentiment.	Polarity scores are very static and absolute, compared to VADER which is calculated by compounded scores.	Need license

2.3. Justify why the selected tool is suitable

VADER was chosen as my primary choice because VADER is a sentiment analysis lexicon validated by humans, which earned them a gold standard. VADER is also suitable and attuned to the context of social media, which I find it is more suitable to perform a sentiment analysis on my data scraped from Twitter. Other than this, I found out that VADER's speed is faster than using Textblob. I did not choose RapidMiner although the graphical user interface is nicer because i am more confident in using Python. When comparing VADER and Textblob, I find out that VADER is more accurate than Textblob.

3. Methodology

3.1. Description of dataset

The dataset I was using was crawled using Twitter developer API by my teammate Jia Cheng.

The dataset details:

- Crawled by Jia Cheng using tweepy.
- Contains 12500 rows of data
- The data is within 7-Sept-2020 9:20pm to 10:40pm.
- The search words to crawl the data are 'Trump' OR 'realDonaldTrump' OR 'DonaldTrump'.

This first version dataset contains 13 columns which are :

Username	- Twitter User's Screen Name
accDesc	- Description of the account
Location	- Where is user tweeting from
Following	- Number of others users that user is following
Followers	- Number of others users who are following this user
Totaltweet	- Total number of tweets by the user
Usercreated	- when the user account was created
Tweetcreated	- when the tweet was created
Retweetcount	- Number of the retweet
favoriteCount	- Total number of tweets that is favorited by user
Text	- full text of tweet
Tweetsource	- Source from user
Hashtags	- Hashtags in the tweet

username	accDesc	location	following	followers	totaltweet	usercreated	tweetcreated	retweetco	favoriteco	text	tweetsour	hashtags
ivory_butkus		California,	17	4	402	17/01/2018 4:40	07/09/2020 9:33	0	0	@the_resi	Twitter fo	[]
ergoking	Poet, stor	Reiberten	2850	1597	35481	23/12/2013 6:25	07/09/2020 9:33	0	0	@morgfai	Twitter fo	[]
stannemei	Photograp	Gilford, NI	1647	98	1678	26/04/2014 12:29	07/09/2020 9:33	0	0	@realDon	Twitter fo	[]
FiddlesTer	Back to th	Kingston-u	5001	4018	5543	04/10/2019 7:56	07/09/2020 9:33	0	0	@GOPCha	Twitter fo	[]

Diagram 3.1.1 Sample Data 1

Second version Dataset:

This dataset only contains 12443 rows and 1 column which is text, that shows users' tweet with full text. All unnecessary columns from the first version dataset set. The duplicate tweet text and remove it because it will affect the sentiment analysis.

text
@the_resistor @DonaldJTrumpJr @realDonaldTrump She is a bot
@morgfair Trump has been pocketing some of it, and they have been paying the Trump children's significant others with campaign cash.
@realDonaldTrump This is what a loser looks like https://t.co/X7IihFNE7v
@GOPChairwoman You have this arse about face, Ms Romney, and not for the first time. Trump is maliciously and mendaciously hinting that a vaccine will be ava
Police Won't Name Leftist Who Attempted to Kill Trump Supporter on Saturday, Girlfriend Says 'This Was Absolutely a Targeted Attack' https://t.co/YK5E7ITlsE via
This was talked about earlier in the year and I knew it was the Trump Card. The DemoRats are emotionally uncontrolled illogical creatures. I am crazy, but I have a

Diagram 3.1.2 Sample Data 2

3.2. Applications of the algorithm(s)

3.2.1 Text PreProcessing

3.2.1.1 Clean text from noises such as @mentions / #hashtag / RT / hyperlink using REGEX

2.1 Clean text from noises such as @mentions / #hastag / RT / hyperlink using REGEX

```

#create a function to clean tweets
def cleanTXT(text):

    text = re.sub(r'@[A-Za-z0-9]+', '',text) #remove @mentions
    #text = re.sub(r'#', '', text) #remove #hastags
    text = re.sub(r'RT[\s]+', '', text)#remove RT
    text = re.sub(r'https?:\/\/\S+', '', text)# remoeved the hyperlink
    text = re.sub(r'(via+)+', '', text) # remove via

    return text

#clean the text
df['Tweet Text'] = df['Tweet Text'].apply(cleanTXT)

#show clean text
df.head()
```

3.2.1.2 Stopwords removal, words tokenization, text stemming and word detokenization

2.2 Stopwords removal, words tokenization, text stemming and word detokenization

```
# remove stopwords
df['Tweet Text'] = df['Tweet Text'].apply(lambda x: ' '.join([w for w in x.split() if len(w) > 3]))

# tokenized the words

df['token'] = df['Tweet Text'].apply(lambda x: x.split())

tokenized_tweet = df['token']

#stemmer for tokenized tweets

stemmer = PorterStemmer()

#apply stemmer for tokenized tweets
tokenized_tweet = tokenized_tweet.apply(lambda x: [stemmer.stem(i) for i in x])

#detokenizer
for i in range(0, len(tokenized_tweet)):
    tokenized_tweet[i] = TreebankWordDetokenizer().detokenize(tokenized_tweet[i])

#change df[tweettext] to processed words
df['Token Text'] = tokenized_tweet
```

3.2.2 Perform Sentiment Analysis using VADER

3.2.2.1 declare tweet_list to receive processed text from dataframe

3.1 declare tweet_list to receive processed text from dataframe

```
#declare list as long as tweet_list
tweet_list = df['Token Text']


#declare sublist as long as tweet_list
sent_list = []
n = len(tweet_list)
for i in range(int(n)):
    k=len(tweet_list)
    sent_list.append(k) # push your entered value

```

```
time: 39 ms
```

3.2.2.2 Return polarity scores fromVADER on list using for loop

3.2 Return polarity scores fromVADER on list using for loop

```
# perform sentimentanalysis on tweet list

for i in range (0, len(tweet_list)):
    sentiment = sid.polarity_scores(tweet_list[i])
    sent_list[i] = sentiment
    print(tweet_list[i])
    print(sent_list[i])
    print ('\n')
```

```
_resistor
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0}


trump been pocket some they have been pay trump children’ signific other with campaign cash.
{'neg': 0.091, 'neu': 0.909, 'pos': 0.0, 'compound': -0.1027}


thi what loser look like
{'neg': 0.382, 'neu': 0.337, 'pos': 0.281, 'compound': -0.2263}
```

3.2.2.3 Move results back into dataframe from returned dictionary

3.3 Move results back into dataframe from returned dictionary

```
#from the dictionary returned from VADER.SENTIMENT, put it in df2 and move it into df1
df2 = pd.DataFrame.from_dict(sent_list)

df2 = df2['compound']

df['Polarity'] = df2

df.head()
```

	Tweet Text	token	Token Text	Polarity
0	_resistor	[_resistor]	_resistor	0.0000
1	Trump been pocketing some they have been paying Trump children's significant others with campaign cash.	[Trump, been, pocketing, some, they, have, been, paying, Trump, children's, significant, others, with, campaign, cash.]	trump been pocket some they have been pay trump children' signific other with campaign cash.	-0.1027
2	This what loser looks like	[This, what, loser, looks, like]	thi what loser look like	-0.2263
3	have this arse about face, Romney, first time. Trump maliciously mendaciously hinting that vaccine will available before election stoke deplorables. Fauci sceptical. Biden Harris call this right necessary.	[have, this, arse, about, face,, Romney,, first, time., Trump, maliciously, mendaciously, hinting, that, vaccine, will, available, before, election, stoke, deplorables., Fauci, sceptical., Biden, Harris, call, this, right, necessary.]	have thi ars about face, romney, first time. trump malici mendaci hint that vaccin will avail befor elect stoke deplorables. fauci sceptical. biden harri call thi right necessary.	-0.2960
4	Police Won't Name Leftist Attempted Kill Trump Supporter Saturday, Girlfriend Says 'This Absolutely Targeted Attack'	[Police, Won't, Name, Leftist, Attempted, Kill, Trump, Supporter, Saturday,, Girlfriend, Says, 'This, Absolutely, Targeted, Attack']	polic won't name leftist attempt kill trump support saturday, girlfriend say 'thi absolut target attack'	-0.7269

3.2.2.4 Compute analysis according to polarity scores

3.4 Compute analysis according to polarity scores

```
# Create a function to compute the negative, neutral and positive analysis
#analysis retrieved from VADER documentation
def getAnalysis(score):
    if score <= -0.05:
        return 'Negative'
    elif score > -0.05 and score <0.05:
        return 'Neutral'
    else:
        return 'Positive'

df['Analysis'] = df['Polarity'].apply(getAnalysis)

# Show the dataframe
df.head()
```

	Tweet Text	token	Token Text	Polarity	Analysis
0	_resistor	[_resistor]	_resistor	0.0000	Neutral
1	Trump been pocketing some they have been paying Trump children's significant others with campaign cash.	[Trump, been, pocketing, some, they, have, been, paying, Trump, children's, significant, others, with, campaign, cash.]	trump been pocket some they have been pay trump children' signific other with campaign cash.	-0.1027	Negative

3.2.2.5 Compute subjectivity

3.5 Compute subjectivity

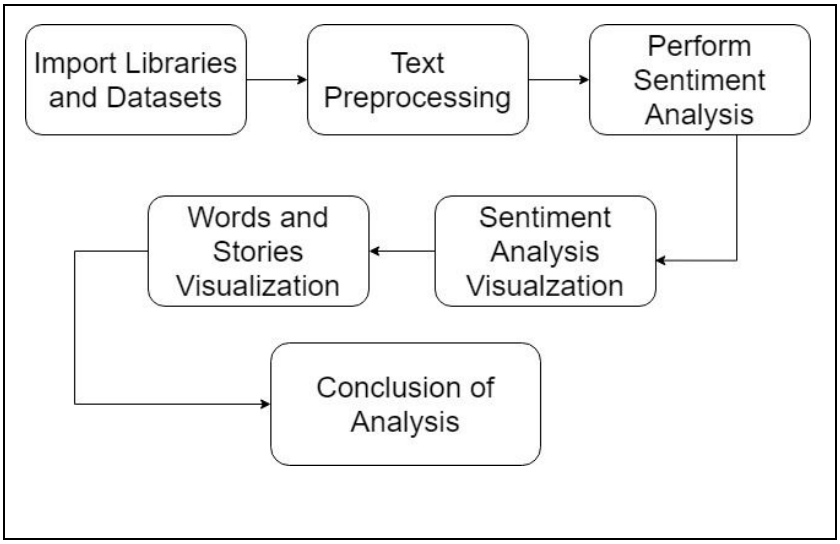
> As VADER has no subjectivity a strongly subjective sentence will have a compound score close to 0, and a neu score close to one. The neu value increases as the proportion of words that match a subjective token from their lexicon decreases. Use the inverse of the neu field of the returned sentiment value (1 - neu). When neu = 1, then subjectivity score is null (0). When neu ~= 0 then you have the highest subjectivity score. Alternatively, you can use the inverse of the absolute value of the compound score (1 - abs(compound)). --by Hiestaa from VADER Github

```
# Create a function to compute subjectivity as VADER has no subjectivity

def getSubjectivity(score):
    sub = (1 - abs(score))
    return sub

df['Subjectivity'] = df['Polarity'].apply(getSubjectivity)
```

3.3. System flowchart/activity diagram



3.4. Proposed test plan/hypothesis

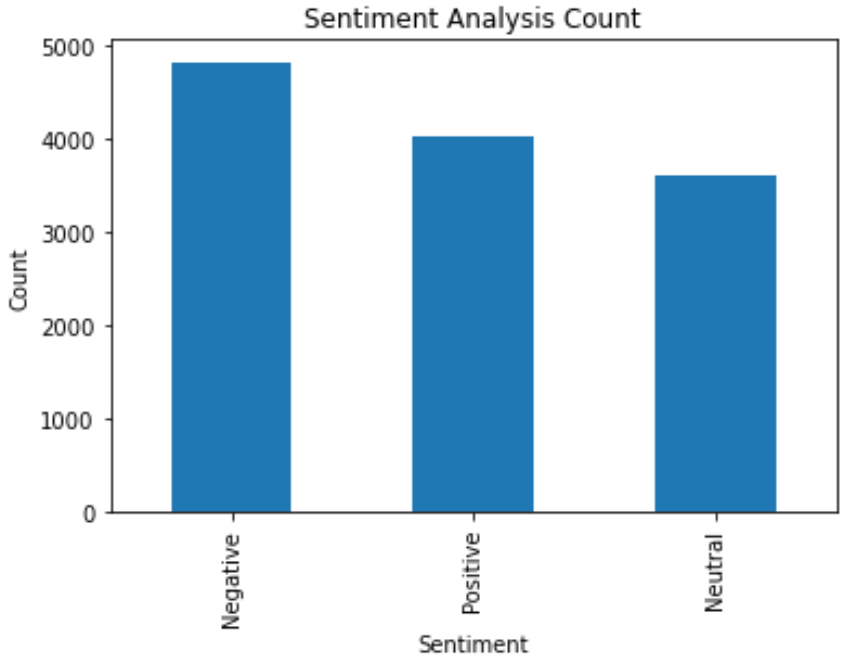
For the hypothesis, I was planning to test the public opinion of tweets about Donald Trump.

The hypothesis are as follows:
H0: Most comments and tweets about Donald Trump are negative.
H1: Most comments and tweets about Donald Trump are positive.

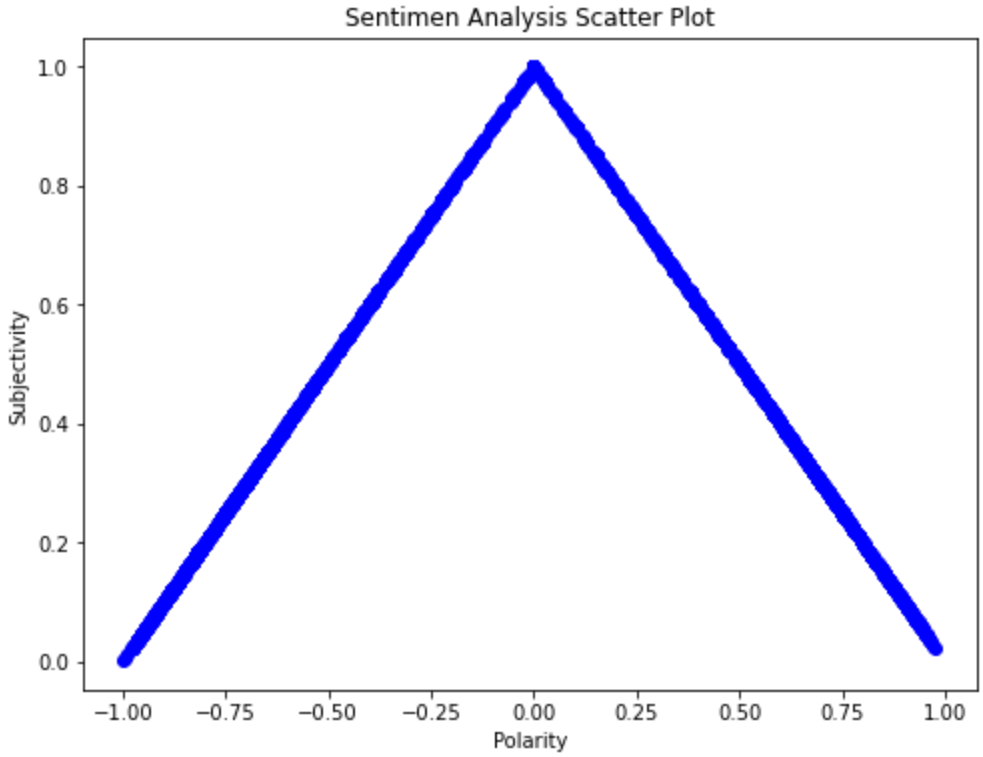
4. Result

4.1. Results

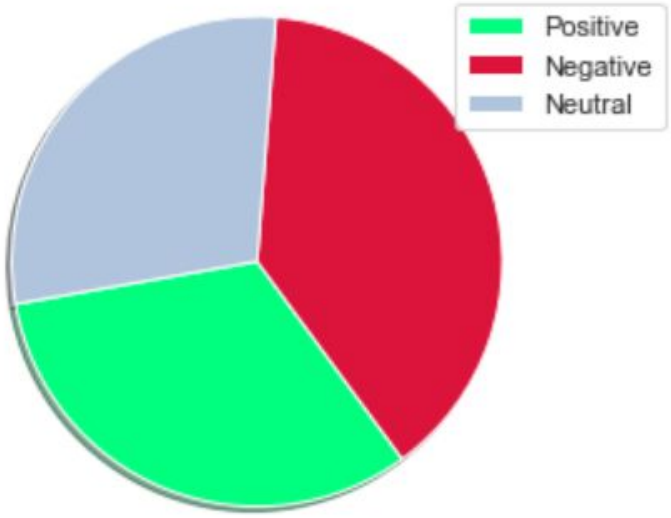
4.1.1 Analysis Count Graph



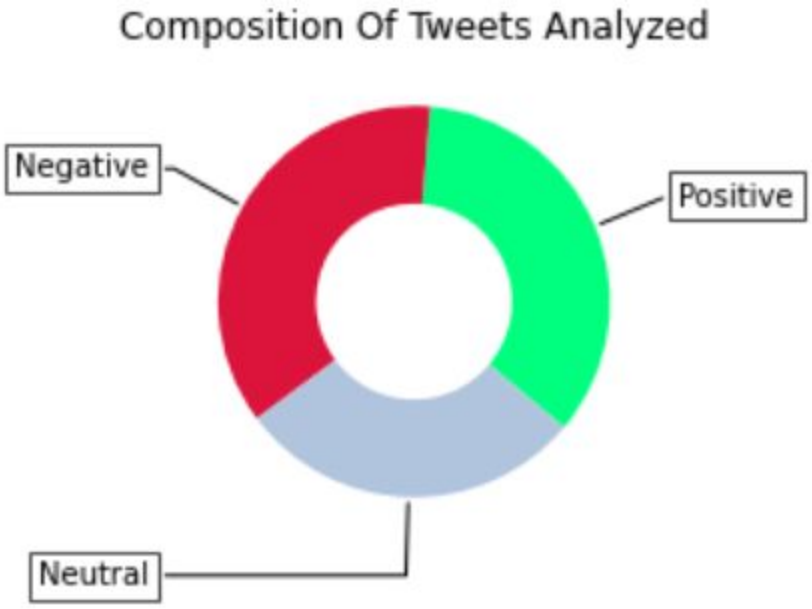
4.1.2 Subjectivity and Polarity Scatter Plot



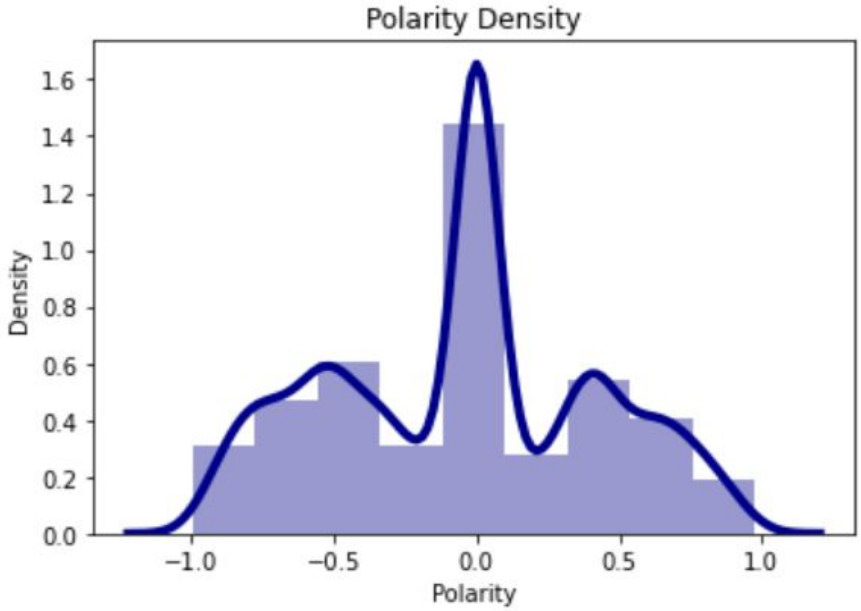
4.1.3 Composition and Piechart of Tweets




```
Total Positive Tweets Percentage:  32.3 %
Total Negative Tweets Percentage:  38.7 %
Total Neutral Tweets Percentage:  29.0 %
```

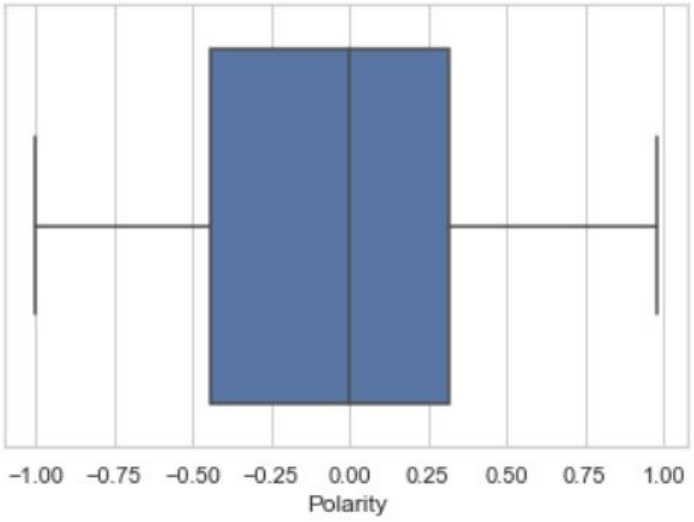


4.1.4 Polarity Density Graph

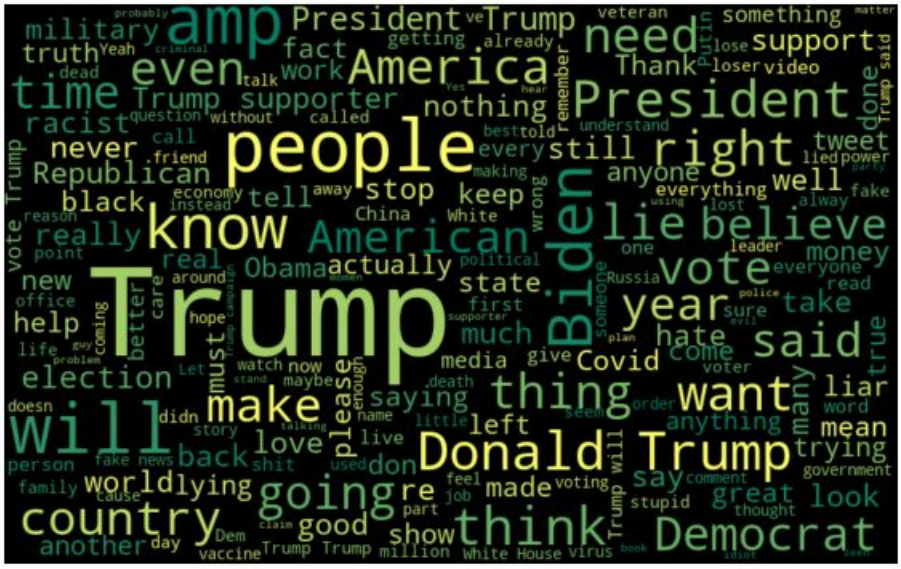


4.1.5 Boxplot For Polarity

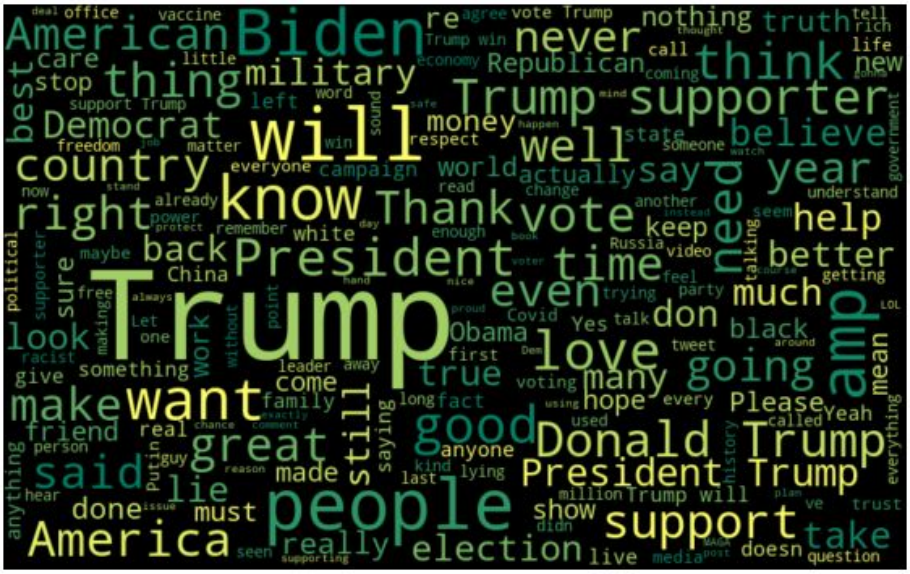
```
count    12443.000000
mean        -0.047177
std          0.468954
min         -0.997500
25%         -0.440400
50%          0.000000
75%          0.318200
max          0.977500
Name: Polarity, dtype: float64
```



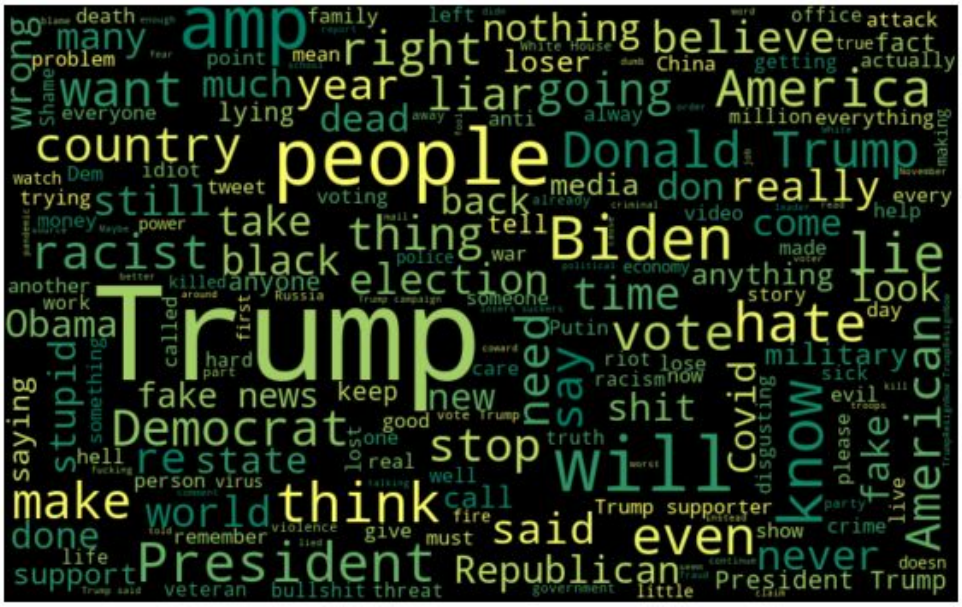
4.1.6 Top Words Use In All Tweets



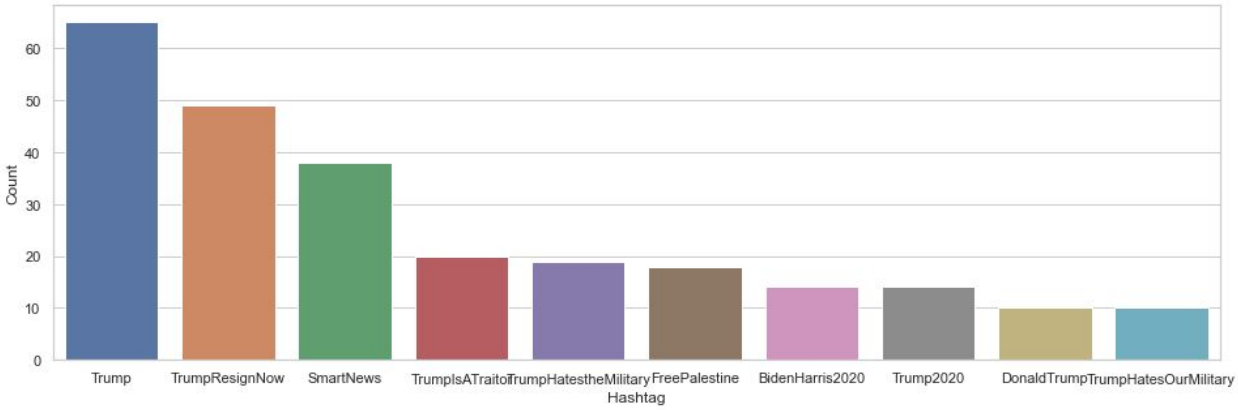
4.1.7 Top Words Used in Positive Tweets



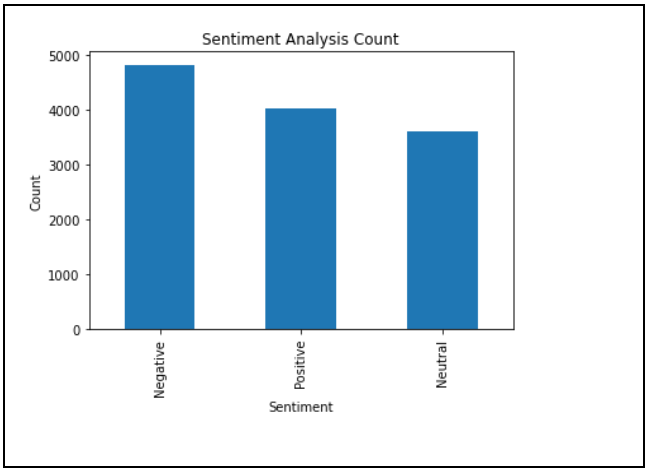
4.1.8 Top Words Used in Negative Tweets

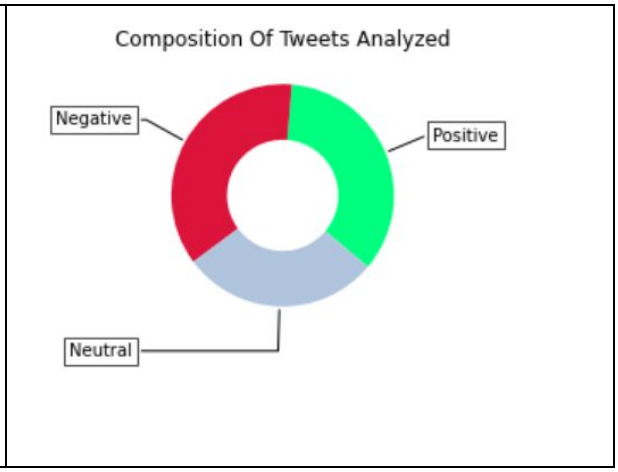


4.1.8 Frequency Plot For Top 10 Negative Hashtags

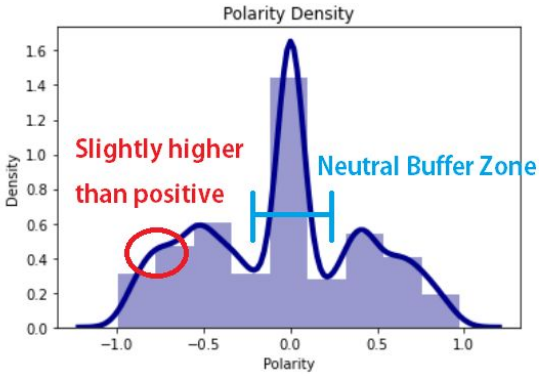


4.2. Discussion/Interpretation

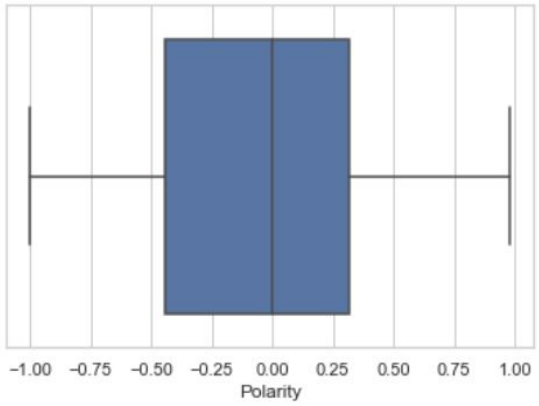




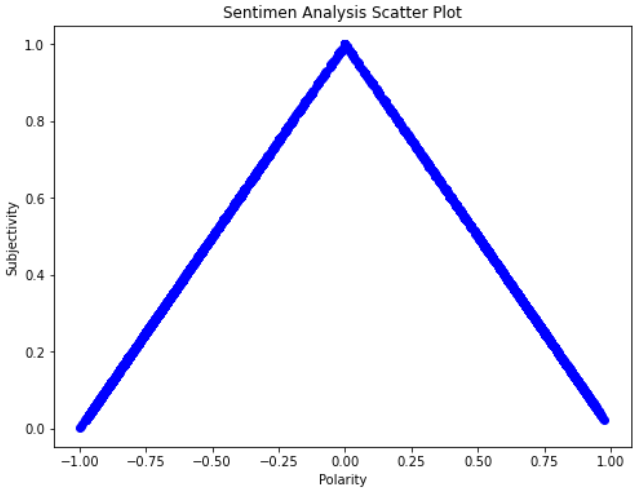
From the Analysis Count Histogram above, we can conclude that negative sentiment has higher count than positive count with the number of 4818 for negative and 4020 for positive. Total positive tweets percentage is 32.3%, while negative and neutral percentages are 38.7% and 29% respectively.




```
count    12443.000000
mean        -0.047177
std          0.468954
min         -0.997500
25%         -0.440400
50%          0.000000
75%          0.318200
max          0.977500
Name: Polarity, dtype: float64
```

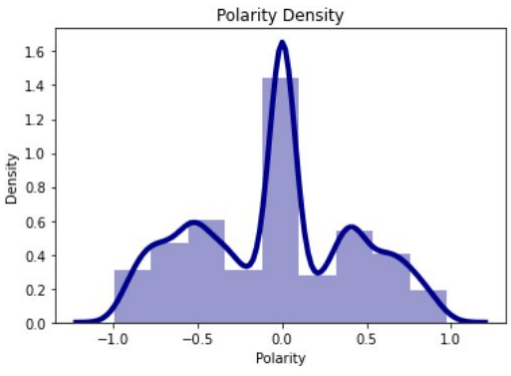


Looking at the polarity density graph, we can see the polarity of negative(<-0.5) is slightly higher than positive (>0.5). VADER neutral polarity has a higher buffer which is from -0.5 to 0.5.From the boxplot graph, we can also see that the minimum polarity is -0.9975 while maximum is 0.9775. The mean of the polarity was 0.04718, therefore we can conclude that there are slightly more negative sentiments compared to positive sentiments.

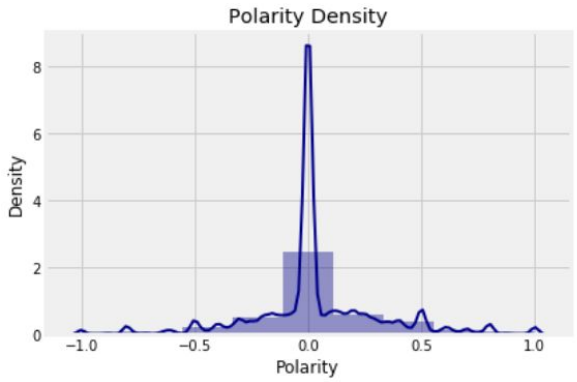


As for the subjectivity and polarity scatter plot, VADER will not return subjectivity because of the properties of its model. By studying the conference paper of VADER, strongly subjective sentences will have a polarity compound score close to 0 and neutral tweets will have a compound score near to 1. The neutral value increases as the proportion of words that match a subjective token from their lexicon decrease. By using the inverse of the neutral field, the subjectivity value can be obtained by (1 - neutral polarity score). When neutral = 1 the subjectivity will be = 0, while when neutral ~= 0 then you have highest subjectivity. Therefore the scatter plot is in linear form. The subjectivity and polarity is linearly related.

Comparing the results with Textblob Library



VADER



TEXTBLOB

By comparing the both results of the same datasets, on the left we are using VADER and on the right we are using TextBlob, it is very obvious that Textblob sentiment analyzer is not sensitive enough, most of the tweets were classified as neutral.

5. Discussion and Conclusion

5.1. Achievements

For the datasets crawled from Twitter, VADER sentiment analysis shows that total negative tweets is higher than total positive tweets. While neutral tweets are the lowest.

Total Negative Tweets Percentage: 38.7 %

Total Positive Tweets Percentage: 32.3 %

Total Neutral Tweets Percentage: 29.0 %

Trump has higher negative tweets. Therefore, we can assume that Trump will be facing a hard time winning this reelection. Donald Trump also does not perform well in his current term.

As for the hypothesis, we fail to reject that Donald Trump has the higher negative tweets compared to positive tweets. All of the objectives for this research were achieved.

5.2. Limitations and Future Works

As we are using datasets crawled from a certain time period, we can only assume the public sentiment that the certain time period. For our case, the time period is 1 hour and 20 minutes. It will be highly inaccurate to use a dataset from such a short time to represent the whole population. A bigger sampling of datasets should be done at a wider spectrum.
If we are to conduct a sentiment analysis on a wider spectrum, we will need to stream the real-time tweets. Computing power and storage resources will be the biggest concern here as the data will be very large in volume.

Other than that, the limitation is that we are using a model built by someone else. We cannot determine the accuracy of the predicted outcome as we have no ground truth. In order to have a better and accurate prediction, we can build a model and fine tuned it to the sentiment analysis on politics. ‘One size fits all’ rules does not work here, it will make the results inaccurate.

Reference & Source

C.J Hutto & Gilbert, E 2014, VADER: A Parsimonios Rule-based Model For Sentiment Analysis of Social Media Text, *Eighth International Conference on Weblogs and Social Media (ICWSM-14)*, Ann Arbor, MI, June 2014.

Hutto 2014, *Vader Sentiment*, viewed on July 28 2020, <https://github.com/cjhutto/vaderSentiment>.

Agrawa, S, 2020, *Sentiment Analysis using LSTM Step-by-Step*, viewed on 10 August 2020, <https://towardsdatascience.com/sentiment-analysis-using-lstm-step-by-step-50d074f09948>

VanderPlas, J, 2017, *Python Data Science Handbook: Essential Tools for Working with Data*, O'Reilly, USA.

Lubanovic, B, 2015, *Introducing Python*, O'Reilly, USA

Smola, A & Vishwanathan, S.V.V, 2008, *Introduction to Machine Learning*, Cambridge University Press, UK.

Shankhdhar Gaurav, 2019, *Sentiment Analysis Methodology*, viewed on 12 August 2020, <https://www.edureka.co/blog/sentiment-analysis-methodology/>.