**STEPHEN OMONDI**

**2019-1002 IST 736 Text Mining**

**TWEETS FOR SENTIMENTS**

**INTRODUCTION**

Presidential debates are important in any mature democracy – ours, perhaps even more important because America is inarguably the epicenter of Western Democracy. Thus, on September 13th, 2019, the Democratic presidential hopefuls gathered to exercise a time-honored tradition of open debate among candidates. Debates like these give citizens and voters a chance to peer into the policies, beliefs, approaches and opinions of those to whom they will choose to vote into the highest office in the land. Debates, like these, also provide a balanced platform for competition of ideas amongst presidential hopefuls. Thus, it is not surprising that presidential candidates see debates as opportunities to deliver winning messages that speaks to the hearts and minds of the electorate – the people.

Consequently, national polls ensue in the days and weeks after presidential debates as candidates rush to measure public post-debate sentiment. The Political candidates rely on this feedback to help them calibrate their messaging to better connect with their constituents in the remaining days to the election day. Meanwhile, a lot of post-debate chatter from the public is generated in offline conversations but also online in social media outlets such as twitter, Instagram, Facebook and similar platforms. It is noteworthy that since the election of the 45th president of the United States, who popularized the messaging platform, Twitter posts have become a leading medium for political chatter.

After presidential debates, Twitter posts (*tweets)* are loaded with public sentiment on policy issues, opinions, support and criticism of the positions taken by candidates. Furthermore, *tweets* are rich in ancillary data such as location where tweets are coming from and time when *tweets* were made which provide rich context for political dialogue. But, can the unstructured data from social media *tweets*, inform on a candidate’s popularity – or lack thereof? Can it be useful in helping presidential candidates understand whether public sentiment is positive, negative or neutral on the issues the politicians work so hard to articulate during debates?

**METHOD**

Sentiment analysis of unstructured data begins with collection of the data itself. Since the overarching structure of Twitter data is through keywords (hastags), it is possible to find all the *tweets* with a common hashtag. For example, #TusliGabbard is used to tag *tweets* pertaining to congress woman Tulsi Gabbard, who is one of the youngest presidential hopefuls in the last Democratic Party debate line-up. A *tweet* is said to be trending when majority of users in the *twitterverse* are tagging a specific hashtag in their *tweets*. Harnessing these *tweets* begins with registering for a developer’s account from Twitter to obtain consumer and access keys which are then used to request data from Twitter.

**Data Collection**

This example uses *#TulsiGabbard* as the keyword and retrieves 1,000 tweets since October 15, 2019. This represents a sample of 10 days of public opinion expressed on Twitter targeting Rep. Tulsi Gabbard. Only original tweets have been kept, with retweets filtered out to give an original account of sentiments.

Figure : How Tweets are built.

Register a Tweeter Developer Account

Create a Tweeter API, obtain consumer and access keys

Connect to Tweeter using the unique API above to download relevant keyword or user data

A Bag of raw, unprocessed tweets from *twitterverse* is generated, and stored into a .csv file for processing and analysis.

The code below connects to Twitter and gathers the tweets:

# pass twitter credentials to tweepy

auth = tw.OAuthHandler(consumer\_key, consumer\_secret)

auth.set\_access\_token(access\_key, access\_secret)

api = tw.API(auth, wait\_on\_rate\_limit=True)

# Search twitter for tweets

# Begin by defining the search term and the date\_since for start date

search\_words = "#tulsigabbard" + "-filter:retweets" # I don't want retweets

date\_since = "2019-10-16"

# collect tweets

tweets = tw.Cursor(api.search,

q=search\_words,

lang="en",

since=date\_since).items(1000) # collects 1000 tweets since date\_since

The resulting collection of tweets was saved into a .csv file for later processing. Raw Tweets are often hard to process without methodical cleaning to separate noise from actual textual data. The cleaning process is usually iterative and is deemed complete only where a corpus of coherent text is arrived at.

Below is a sample of the raw tweets before cleaning and processing.

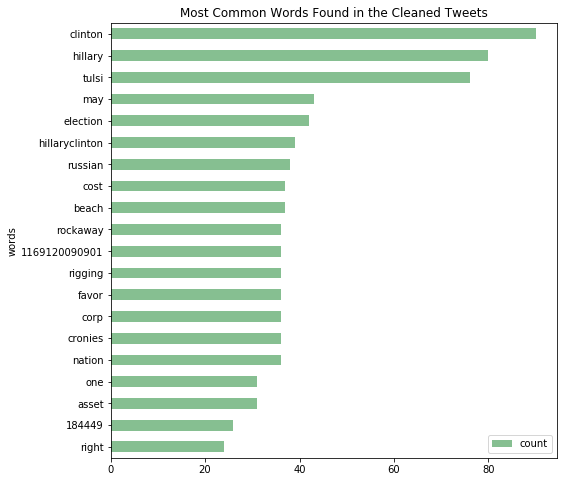
[['Zurchmom', 'DNA USA & Beyond', datetime.datetime(2019, 10, 5, 3, 9, 35), 'NEVER FORGET #TULSIGABBARD IS A #DEMOCRAT WITH A LIBERAL AGENDA. SHE IS OWNED &amp; CONTROLLED BY THE DEMOCRAT PARTY &amp;… https://t.co/hXxe07aEvy'], ['keny\_berd', 'USA', datetime.datetime(2019, 10, 14, 22, 10, 8), '#TulsiGabbard now on #Hannity wearing a white suit! What is that all about? https://t.co/303yZIpYsY https://t.co/fbTGM7uLTU'], ['sianetta', 'Proud Clevelander/Naples-ite', datetime.datetime(2019, 10, 19, 11, 22, 33), '@seanhannity For some reason you felt compelled to let #TulsiGabbard come on your show&amp;pander to ur audience attemp… https://t.co/Py7gnAVXrE'], ['keemau', 'So. Florida', datetime.datetime(2019, 5, 22, 3, 42, 48), 'Finally someone standing up to Clinton. I just made my first political contribution in in 45 years. Stay strong sol… https://t.co/RsI3vvBFHt'], ['hhanzosteel', 'Mikawa Province', datetime.datetime(2019, 10, 31, 20, 52, 32), 'Hey #Tulski #TulsiGabbard @TulsiGabbard do you have any comments about this? #NotYouTulsi https://t.co/LNVvBBRcC2']]

**Data Cleaning**

The following actions were taken in order to make the body of tweets coherent and ready for further analysis:

1. Converted the tweets to lowercase to establish uniformity. This helps reduce vocabulary generated since words are treated similarly regardless of case difference. For example, an uppercase “Boy”, is considered similar to a lowercase “boy.”
2. Removed Stop words using the English dictionary. Stop words are filler words that are used to connect the critical vocabulary in a body of text, such as conjunctions, prepositions and articles e.g. it, is, for, in, at, on, at, etc.
3. Removed leading and ending space found in the words and sentences.
4. Removed hashtag symbol (#) in the body of text – this eliminated other hashtags used in the tweets
5. Removed weblinks which typically come with tweets
6. Removed all non-ASCII patterns
7. Removed words with less than 3 letters

Using basic split of words in the cleaned text above yielded **6,256** vocabularies, following of which had the most common word frequencies:



**ANALYSIS**

**Tokenization**

Tokenization helps in building the vocabulary base from the body of cleaned *tweets*. This is achieved by splitting the text into its constituent singular word parts or tokens. Python NLTK Library was used to achieve this type of tokenization. A total of **6,264** unique tokens were found in the cleaned tweet.

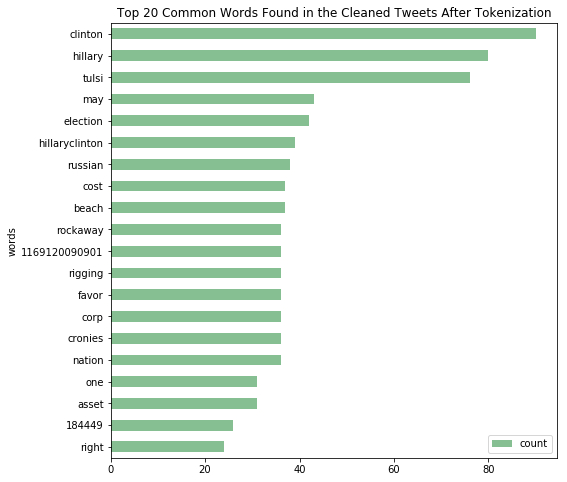
####Tokenize######

tokens = nltk.word\_tokenize(str(df))

tags = nltk.pos\_tag(tokens)

print(len(tokens))

The chart below shows the top 20 most common words/tokens after cleaning the collected tweets.



**Removing Additional Stop words**

Additional stop words were identified and removed by first updating the stop words list to include the added items. In the end, the vocabulary count reduced to **6,098** tokens.

####Remove Additional Stop Words######

stopwords.update(["","tulsi","tulsi2020", "1169120090901", "184449", "//", '"', "'", "\*", ":", ";", "?", "@", "https","``","s","t","[","]"])

def removeStopWords(wordlist, stopwords):

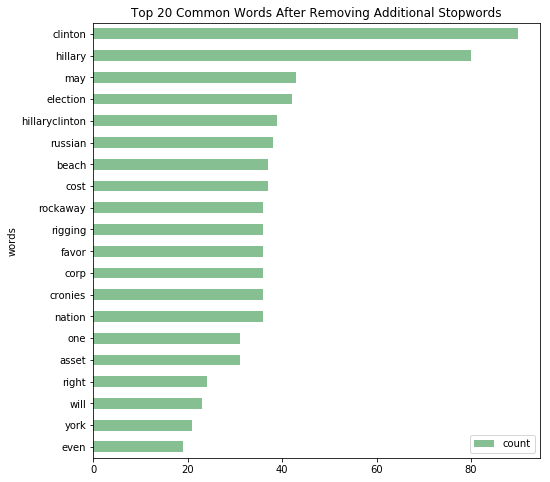
return [w for w in wordlist if w not in stopwords]

words = removeStopWords(tokens, stopwords)

#print(words)

print(len(words))

A display of top 20 most frequently used words after removing additional stop words such as the hashtag used for tweet search (tulsi2020) and the name of the candidate (tulsi).



**Stemming**

Reducing words to their root/stems (stemming) can also reduce the vocabulary size. There were **6,098** words after stemming indicating that stemming had no impact on the size of the vocabulary even though the structure of the vocabulary changed as shown below:

######################Stemming

ps = PorterStemmer()

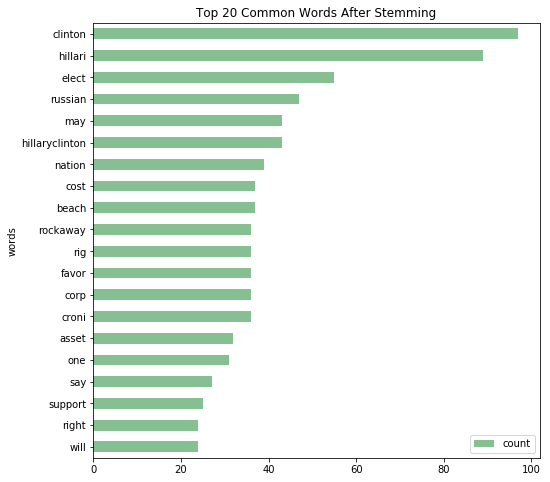
stem\_words = []

for word in words:

stem\_words.append(ps.stem(word))

print(ps.stem(word))

len(stem\_words)



**RESULTS**

Using the cleaned tweets from previous sections generates an informing word cloud of vocabularies used when people are talking about Rep Tulsi Gabbard on Twitter.

**Word Cloud**

Using all words in the vocabulary to generate a word cloud:

wordcloud = WordCloud(

background\_color ='azure',

stopwords = stopwords,

max\_words = len(stem\_words),

min\_font\_size = 10).generate(str(stem\_words))

# display wordcloud

plt.imshow(wordcloud, interpolation='bilinear')

plt.axis("off")

plt.show()

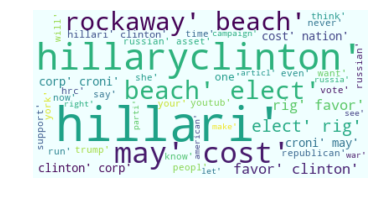


Figure : Word cloud of what people say about Rep Tulsi

Word cloud using top 100 words only:

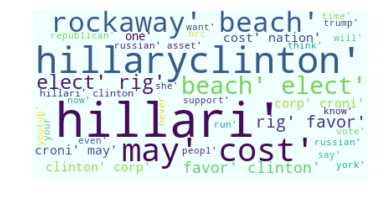


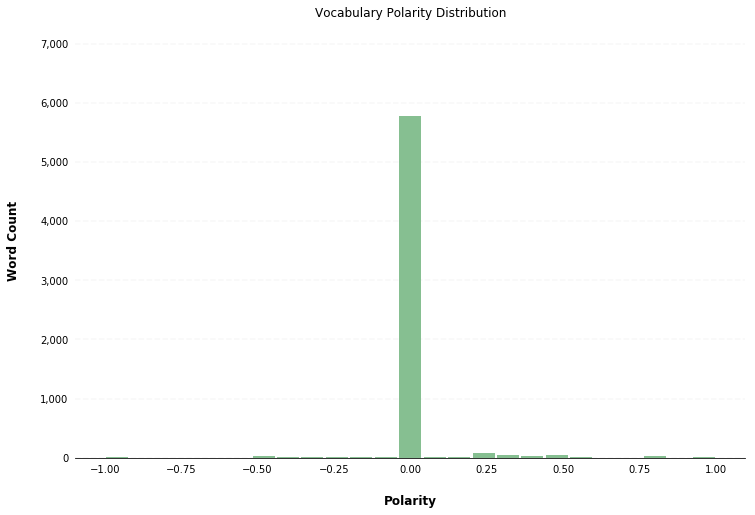
Figure Word Cloud of top 100 most frequently used words

When all words are taken into the word cloud, the terms “*rockaway”, “beach”, “hilaryclinton”, “hillari”, “may”, “cost”, “rig”, “favor”* are used more frequently than other words. This pattern repeats even when a subset consisting of top 100 tweet vocabularies are used.

“Clinton” and “Hillary” appear at the very top of the most frequently used words in these tweets. Other notable words are *Russian, may, nation, will, cost, favor, rig, say, one, run, time, support* and *asset*.

**Mining Sentiments from the Tweets**

So, what do these vocabularies in tweets say about how people think of Rep Tulsi Gabbard? Sentiments from the vocabularies were elicited using the *TextBlob* machine learning algorithm. The sentiment property of in *TextBlob* returns a named tuple of the form *Sentiment (polarity, subjectivity).* The polarity score is a float ranging from [-1.0, 1.0] where -1.0 is very negative, and 1.0 is very positive. The subjectivity is a float within the range of [0.0,1.0] where 0.0 is very objective and 1.0 is very subjective. The histogram below shows the polarity scale of vocabularies in the tweets.



It is evident from the distribution above that word tokens used in #tulsigabbard are vastly neutral, with very small amount being very negative or very positive. It is deducible that a net neutral sentiment is obviously the twitterverse response to #tulsigabbard presidential candidacy. However, very negative words used to describe #tulsigabbard should draw the attention of this presidential candidate, such as the words shown below.

**Top 10 Very Negative Words in #tulsigabbard**

|  |  |
| --- | --- |
| **Word** | **Polarity** |
| Worst | -1.0 |
| Evil | -1.0 |
| Base | -0.8 |
| Hate | -0.8 |
| Crap | -0.8 |
| Fake | -0.5 |
| Corrupt | -0.5 |
| Wrong | -0.5 |
| Unfair | -0.5 |
| Meaningless | -0.5 |

Similarly, very positive words used in #tulsigabbard should reassure this presidential candidate that her messaging is well received, such as the words shown below:

**Very Positive Words used in #tulsigabbard**

|  |  |
| --- | --- |
| **Word** | **Polarity** |
| Sure | 0.5 |
| Love | 0.5 |
| Better | 0.5 |
| Latest | 0.5 |
| Popular | 0.6 |
| Nice | 0.6 |
| Fair | 0.7 |
| Great | 0.8 |
| Elect | 0.8 |
| Brilliant | 0.9 |
| Best | 1.0 |

**CONCLUSION**

#tulsigabbard has been used frequently in tweets referring to Rep Tulsi Gabbard since the second democratic debate to characterize people’s sentiments. It is evident that whereas the last Democratic Presidential Candidate – Sec Hilary Clinton – is not running for office, her name is frequently mentioned when people talk of Rep Gabbard. Rep Gabbard is also linked to Russia, and it also appears that she is being linked to words such as rigging, favor, and corporation.

While a more complete sentiment analysis is warranted to truly understand the public sentiment on Rep Gabbard, the top 20 words used suggests an overarching linkage to ideas of rigging, Russia, corporations and favor. These words reflect fear eminent in Secretary Clinton’s own assertions about such things as Russian influence and election meddling using Rep Tulsi Gabbard as conduit. It is imperative upon Rep Tulsi Gabbard to delink and debunk from these ideas to sway the public narrative towards more favorable words. However, it is also reassuring to know that there are very positive words being used to describe her as a presidential candidate. While there fewer positive and negative words used to describe her candidacy, there is overwhelming neutrality of opinion about Rep Tulsi Gabbard.