# Tweet-Mining 2016 U.S Presidential Candidates

## Addressing the Elephant in the Room

**Group Members**: Patrick Cane, Lani Haque, Chris Shoniker, Touba Warsi

STAT 5703 Research Assignment

December 1, 2015

**Statement of Duties**:

Patrick Cane:  Data collection, KNN Classification and relevant code

Lani Haque:  Literature review, write up, presentation creation

Chris Shoniker:  Sentiment analysis method and relevant code

Touba Warsi:  Literature review, write up, presentation creation

# Table of Contents

**Abstract** ... 4

**Introduction** ... 5

Text Mining ... 5

Classification using K-Nearest Neighbours ... 7

Sentiment Analysis ... 8

Twitter ... 9

**Methods** ... 10

Data Collection ... 10

Classification using K-Nearest Neighbours ... 11

Data Cleaning ... 11

Data Pre-Processing ... 12

Model Building ... 13

Sentiment Analysis ... 14

Data Cleaning ... 14

Scoring Functions ... 15

**Results** ... 15

Classification using K-Nearest Neighbours ... 15

Sentiment Analysis ... 17

**Discussion** ... 22

**References** ... 24

**Appendix A -** Tables and Figures ... 26

**Appendix B -** R-Code ... 37

# Abstract

The main goal of text mining is to glean relevant information from text data, with the biggest hurdle being the ability to transform text data into a format easily analyzed by software or computer programs. For the purposes of this report, we focused on using Twitter data from five 2016 US presidential candidates to first create a classification model using the k-Nearest Neighbours (KNN) algorithm to identify politicians based on tweets as well as creating a classification model to identify party affiliation based on tweets. Secondly, we performed sentiment analysis on the tweets to determine the overall emotional polarity of each politician using a vote count of positive and negative words to score each tweet. We chose to analyze 1000 tweets each from the following US presidential candidates: Jeb Bush, Hillary Clinton, Ted Cruz, Bernie Sanders and Donald Trump. Interestingly, the models we created using KNN yielded an accuracy of 66% when classifying politicians and an accuracy of 84% when classifying party affiliation. Due the inherent noise within Twitter data, these results met expectations. Furthermore, our sentiment analysis revealed a fairly uniform emotional response across politicians, with all of them projecting a neutral and joyous disposition, instead of appearing overly negative, overly positive or depicting extreme emotions such as fear, disgust or sadness.

# Introduction

## Text Mining

Text mining is similar to data mining except now the data source is a text document, unstructured or structured. The idea behind text mining is to extract useful information from a collection of documents. Text mining is an interdisciplinary field including methods and theory from other fields such as information retrieval, machine learning, statistics, computational linguistics and data mining. [9] In figure 1 we can see that the applications of text mining are fairly broad due to the wide variety of fields that it covers. Some of the sectors where text mining can and is being used are in publishing and media, telecommunications technology, insurance and financial markets, among others. Gupta and Lehal provide an overview of text mining, techniques and its applications. [7] The value of text mining and the information it can extract and offer is being recognized in other areas as well such as market analysis. More businesses are using text mining to analyze competitors, monitor customer opinions and more with the aim of understanding the market and their position in the market better. [7]

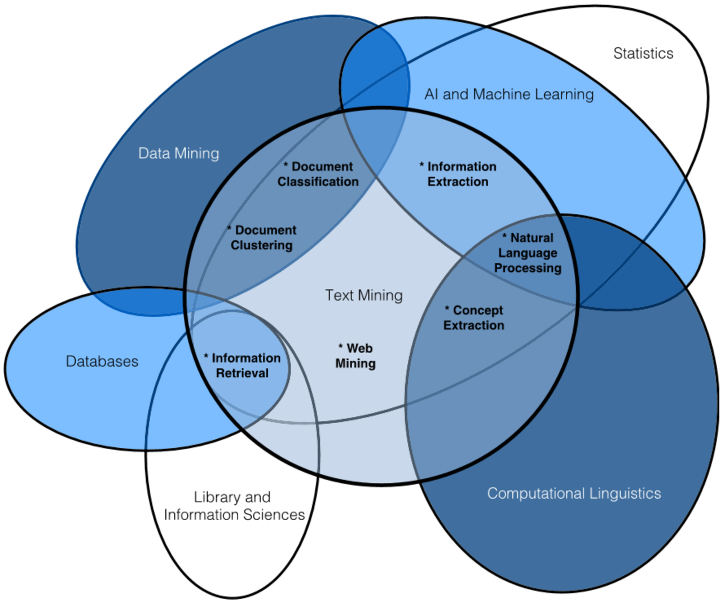


*Figure 1 Venn diagram of text mining and its intersection with other relevant disciplines.[13]*

Social media sites like Facebook, Twitter, and Google+ are being used more by businesses for market analysis. Traditional methods for data collections for the purposes of market analysis, such as focus groups or face to face interviewing are becoming more costly, increasing in non-response rates and are more time consuming. As opposed to traditional methods of collecting consumer opinions, more and more people are expressing their views and opinions on social media platforms making this information more easily accessible and free for businesses. As a result, more large businesses and companies are creating an online presence on social media platforms. The main dataset in text mining on which analyses are performed is called a "Corpus," which is a large structure for text data.

Overall, a brief survey of text mining and its application can be found in the papers by Hotho et al. and Gupta and Lehal. [7,9], both consider the techniques and methods used in text mining as well as applications of text mining. The latter, a more recent paper, considers techniques such as topic tracking, categorization and information visualization among others while the former goes into more detail on methods such as Naïve Bayes classifier, nearest neighbour classifier, decision trees, support vector machines and more. Text mining for the purpose of sentiment analysis is considered in a wide range of industries including insurance [8], food services such as the pizza industry, [15], and large consumer brands such as IBM, Nokia and DHL [15]. All three used social media sites, including Twitter, from which to collect their data for purposes such as sentiment analysis. Mostafa [15] mentions further research in the direction of sentiment topic recognition. This is an interesting extension considering not just the overall sentiment of the text but the most representative topic behind the sentiment. Elkan, [3], goes into the more detail on methods used when considering topic models, e.g. generative processes and training vs Gibbs sampling. Bermingham and Smeaton [2] consider microblogging. Their results were positive and encouraging for sentiment analysis in microblogs however, they also noted that what made microblogs "noisy", the punctuation etc., was in fact beneficial to the classifiers. In the paper by Zhong et. Al [16], term based methods are considered when text mining. A major advantage includes efficient computation performance. However, term-based methods suffer from polysemy, multiple meanings for one word and synonymy, multiple words having the same meaning. In the paper by Li et al. [22], pattern-mining techniques are used. Two challenging issues that arise are long patterns are usually specific for the topic but appear in documents with low support. Misinterpretation is also a problem meaning measures such as support and confidence in pattern mining turn out to be unsuitable. In the paper by Luo et al. [1], sentiment analysis is used on comment data from financial message boards to gain knowledge of investors' opinions.

## Classification using K-Nearest Neighbours

The k-nearest neighbour algorithm (KNN) is a non-parametric method. It is used in pattern recognition for classification and regression. In this report and throughout, KNN will be used for classification. In both cases, the input consists of k closest training examples in the feature space and the output depends on whether KNN is used for classification or regression. The output of KNN is a membership to a class. An object is classified by majority vote of its neighbours. K is typically a small integer because even though larger values of k reduce the effect of noise on the classification, the resulting boundaries between classes may become less distinct. There are various heuristic techniques to determine k. When k=1 the object is assigned to the class of the single nearest neighbour. This special case is called the nearest neighbour algorithm. KNN is one of the simplest of all machine learning algorithms. Weight may be assigned to the contributions of the neighbours so that nearer neighbours contribute more to the average than those farther away. A common distance metric used for continuous variables is the Euclidean metric, but other distance measures such as the Manhattan metric or the Infinity Distance metric are available for use. A shortcoming of the KNN is its sensitivity to the local structure of the data. To avoid the effect of the curse of dimensionality, reduction is usually done before applying KNN. The dimensionality curse in KNN means the Euclidean distance is not helpful in high dimensions because all points are almost equidistance from the search query point.

## Sentiment Analysis

Sentiment analysis, or opinion mining, aims to extract subjective information from text documents. Usually the aim in sentiment analysis is to determine the attitude of the author on a particular topic or the overall polarity or subjectivity of the document text exchange. Polarity or subjectivity may be thought of as classifying the overall "feel" of the text as positive, negative or neutral. This now becomes a classification problem into 2 or more classes. Emotion recognition is a further extension of sentiment analysis where the goal is to gain a better understanding of the author's opinion. The classes are further refined from positive, negative and neutral to emotions such as anger, disgust, fear, happiness, sadness and surprise. Usually a dictionary of positive and negative words is used as a reference for sentiment analysis. SentiWorkNet and Hu and Liu's lexicon are examples of frequently used dictionaries. [20] A dictionary in sentiment analysis is a listing of words and the accompanying polarity and/or the emotion they express. In some cases, making your own dictionary or adding to existing dictionaries based on the vocabulary in the desired text to be mined is necessary. Tables 1 and 2 show excerpts of emotion and subjectivity dictionaries, respectively.

| **Term** | **Emotion** |
|---|---|
| wretch | sadness |
| wretched | sadness |
| wroth | anger |
| wrothful | anger |
| yucki | disgust |
| yucky | disgust |
| zeal | joy |
| zealous | joy |

*Table 1 Emotion dictionary*

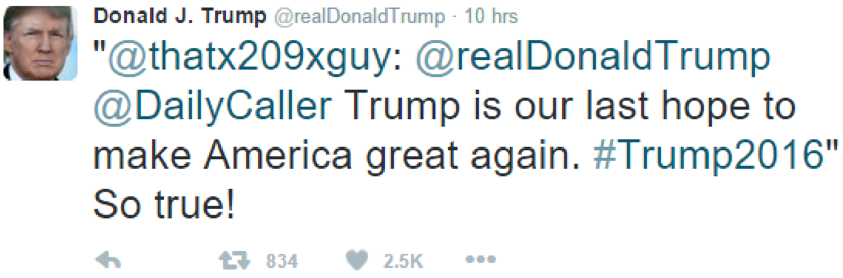
| Term | Subjectivity |
|---|---|
| worsening | negative |
| worship | positive |
| worst | negative |
| worth | positive |
| worthwhile | positive |
| worthiness | positive |
| worthless | negative |
| worthlessly | negative |
| worthlessness | negative |

*Table 2 Subjectivity dictionary*

Sentiment analysis can be used for marketing purposes whether to get a sense of how a company is performing, redesigning marketing and advertising campaigns, or analyzing the competition. Social media communities such as Facebook and Twitter provide a platform for consumers to express their views on events, products and more. Through social media posts, data is generated that can be used to gain potential information about a product or service. As a result, more companies and businesses are creating an online presence on social media sites. Twitter is one particular social media platform that businesses are using to perform sentiment analysis. [8,14]

## Twitter

Twitter is a microblog launched in 2006 where posts or tweets are a maximum of 140 characters in length. All users have a timeline showing all their posts. Users can follow other users on twitter, view their posts, and can retweet, "RT", or quote posts by others onto their timeline. Furthermore, users may send direct messages, "DM", to a user without other users seeing the message. Users may directly address others with the '@' symbol with a message that can be viewed by all. Posts can be tagged with a hashtag '#' about a particular topic determined by the user. Lastly, users may like the posts of others and create lists of favorite posts. With all the functionality available, the Twitter environment can be very "noisy". As a result, the data obtained from this environment can also be very noisy. Figure 2 shows an example of a tweet by Donald Trump.



Donald J. Trump @realDonaldTrump · 10 hrs

"@thatx209xguy: @realDonaldTrump @DailyCaller Trump is our last hope to make America great again. #Trump2016" So true!

*Figure 2* - Example of a tweet by Donald Trump

Not only can businesses benefit from a presence on social media and Twitter, but people, celebrities, and politicians can also use these platforms to express their views, gain a following, and become 'leaders' in a field etc. In particular, politicians can get a sense of how they and their party are doing politically. In this report we text mine to perform sentiment analysis on five 2016 US presidential candidates. We used text-mining to create a classification model using the k-Nearest Neighbours (KNN) algorithm to identify politicians based on tweets and to also create a classification model to separate politicians by party affiliation based on tweets. Secondly, we performed sentiment analysis on the tweets to determine the overall emotional polarity of each politician. We chose to analyze 1000 tweets each from the following US presidential candidates: Jeb Bush, Hillary Clinton, Ted Cruz, Bernie Sanders and Donald Trump.

# Methods

## Data Collection

All data collection procedures and statistical analyses were conducted in the R statistical environment. Tweets from the candidates of interest were obtained from Twitter using the package "twitteR," an R-based Twitter client updated as recently as July 2015. This package allows users to grab subsets of Twitter data for analyses [30].In order to obtain the tweets, a one-time OAuth authentication was required to access Twitter for data collection using R, after which tweets could be obtained by either desired term or by user or both. In order to compile data for each presidential candidate, the most recent 1000 tweets from each politician were collected on 2015-11-27, 20:45. Afterwards, the R-package “tm” was used to compile the twitter data into a “Corpus” for each politician, which is the main structure for managing documents in “tm” [31]. The “tm” package was used extensively to manage, clean and prepare data for further analyses. To begin, the raw text from each tweet was extracted using the “getText” function and compiled into 1000 separate text documents, after which the text documents were collected into a raw-data Corpus for each politician through the use of the “Corpus” function (See Appendix B1 – Data Collection).

## Classification using K-Nearest Neighbours

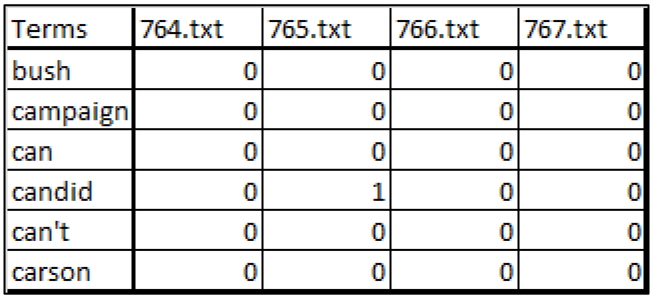
K-Nearest Neighbours (KNN) is a distance-based classification algorithm that aims to classify a new instance based on the known classification of other instances [32]. Given some labelled data points for training and also unlabelled data for testing, the algorithm proceeds by identifying the k nearest labelled data points to the test data using distance measures such as Euclidean distance, Manhattan distance or Infinity distance. Then the classification of the test data is decided by majority vote of the training-set members, with ties broken at random. The algorithm is quite easy to understand, with a minimal training phase and a simplicity that underlies its ability to work well in practice. For the purposes of this report, using KNN yielded high classification accuracy, which will be discussed later, when compared to current literature and as such we decided to proceed with this simple yet effective model.

### Data Cleaning

The data were cleaned in order to ensure that the text data consisted of only words and terms that could be used to identify the politician’s unique style of expressing themselves. Each raw corpus underwent the same cleaning procedure wherein the following were removed: URLs, punctuation, emojis, twitter mentions of other users and self-references (See Appendix B2 – Classification Data Cleaning). Furthermore, the letters in the corpus were changed to all lowercase, and common English stopwords such as “I”, “this”, “a” and “have” were removed, as is common before processing of natural language data [33]. Each cleaned Corpus was then saved separately to allow for further analyses.

## Data Pre-Processing

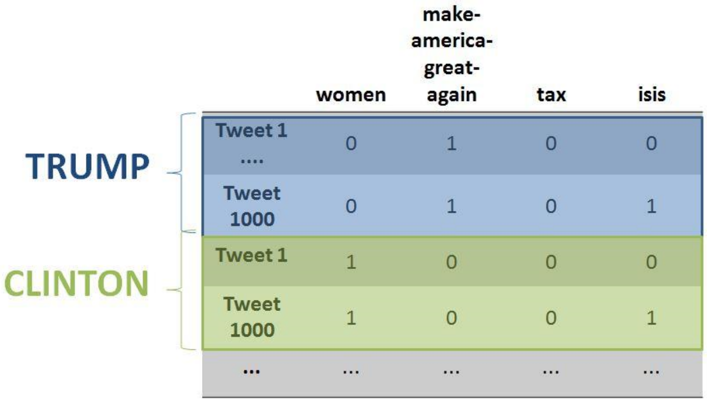
In order to prepare the data for KNN classification, first a “Term Document Matrix” (TDM) was created using the “TermDocumentMatrix” function.  Similar in nature to an incidence matrix, the TDM presents each tweet of a corpus as columns and all of the words in the corpus as rows, with the number of times each word appears in a tweet as an entry in a cell as seen in Figure 3.



| Terms | 764.txt | 765.txt | 766.txt | 767.txt |
|---|---|---|---|---|
| bush | 0 | 0 | 0 | 0 |
| campaign | 0 | 0 | 0 | 0 |
| can | 0 | 0 | 0 | 0 |
| candid | 0 | 1 | 0 | 0 |
| can't | 0 | 0 | 0 | 0 |
| carson | 0 | 0 | 0 | 0 |

*Figure 3* - Sample of Donald Trump’s Term Document Matrix

As one can imagine, it is possible for TDMs to get quite large for even moderately sized data sets; for example Donald Trump’s TDM would have yielded a matrix of more than 1000 rows and 1000 columns.  This is because most words are rarely used, creating a very sparse matrix. Therefore, the function “removeSparseTerms” was used to remove the rarely used words with the intention of improving the computation time and accuracy of the model. It is important to note that it is up to the analyst to choose the tuning parameter that determines the threshold of sparsity at which to remove a word. For the purposes of this report, a sparsity tuning parameter of 0.98 was chosen because it yielded the highest classification accuracy, while a value of 0.97 removed too many terms leading to poor modeling while a value 0.99 did not satisfactorily reduce the dimension of the matrix. Finally, once each politician's TDM was stripped of sparse terms, each TDM was transposed such that words became columns and tweets became rows. All of the transposed matrices were then stacked in to one large incidence matrix. This stacked incidence matrix also contained a separate column indicating which politician was the author of a set of tweets as seen in figure 4. All relevant RCode can be viewed in Appendix B3 – KNN Classification.



| | | women | make-america-great-again | tax | isis |
|---|---|---|---|---|---|
| TRUMP | Tweet 1<br>.... | 0 | 1 | 0 | 0 |
| | Tweet 1000 | 0 | 1 | 0 | 1 |
| CLINTON | Tweet 1 | 1 | 0 | 0 | 0 |
| | Tweet 1000 | 1 | 0 | 0 | 1 |
| | ... | ... | ... | ... | ... |

*Figure 4* - Sample of Stacked Incidence Matrix

## Model Building

In order to proceed with creating a KNN classification model, the stacked incidence matrix was split into a training set of 70% of the data and a test dataset of the remaining 30%. As stated previously, the KNN algorithm is easily employed in R using the "knn" function. Within this function, the analyst simply has to specify the training and test data, and Euclidean distance is used as the distance measure to classify the test data based on majority vote of the training data. For the purposes of this report, we were interested in using the KNN algorithm for a two-fold purpose: 1) to create a model to identify politicians, and 2) to create a model to separate tweets by party affiliation of the politician. In order to identify politicians, the stacked incidence matrix, as mentioned previously, had a column indicating the name of the author of each tweet. And in order to separate politicians based on party affiliation, the stacked incidence matrix instead had a column indicating the party affiliation of the author of each tweet instead of their names. Next, a confusion matrix was created for each model to visualize the performance of the KNN algorithm in classifying each politician correctly or classifying each party member correctly.

## Sentiment Analysis

As mentioned previously, sentiment analysis aims to extract subjective information from text documents.  Usually the aim in sentiment analysis is to determine the attitude of the author on a particular topic. For the purposes of this report, we wanted to perform sentiment analysis on the tweets of each politician to determine the overall emotional polarity of each candidate. We hypothesized that since each candidate was marketing themselves as a “brand” and dealing with sensitive topics in the world of politics, the candidates would each have a fairly neutral emotional polarity so as to appeal to as large a number of constituents as possible.

### Data Cleaning

Given the raw Corpus for each candidate, the data were not cleaned as extensively for sentiment analysis as they were for the previous model building. This is due to a major component of sentiment analysis being the ability to match each word in a tweet to a “dictionary” [35]. Since any words in a tweet that did not match a dictionary entry would be discarded anyways and truncating words might cause them to not match a dictionary entry, we thought it best to keep the data cleaning to a minimum in comparison to the classification. For this reason, the raw Corpuses were first stripped of all punctuation, the text was all changed to lowercase and the “str_split()” function was applied to split the tweets into separate words. This prepared the data to be compared to the subjectivity and emotion dictionaries.

### Scoring Functions

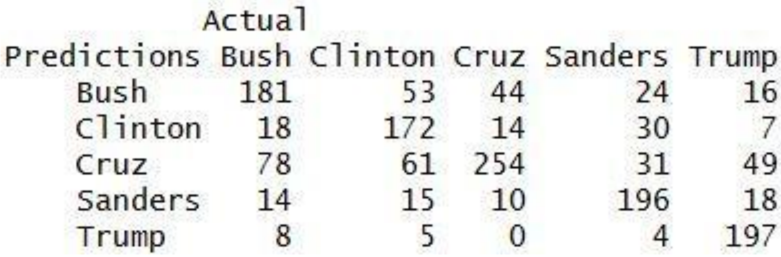
As mentioned previously, sentiment analysis relies on comparing the text of interest to compiled dictionaries containing words and the subjectivity level or emotion attached to those words. In order to proceed with our analyses, the dictionaries acquired from the "sentiment" package were first stripped of duplicate words and divided into separate emotion or subjectivity dictionaries. In this manner, we ended up with two subjectivity dictionaries (positive and negative) and six emotion dictionaries (anger, disgust, fear, joy, sadness, and surprise). Next, scoring functions were created to calculate the subjectivity and emotion of each tweet [34] (See Appendix B4 – Sentiment Analysis).

A few items of note; the Sentiment package has not been updated since 2013 and hence removed from the Cran-R repository. This was not a major limiting factor for the purposes of this report due to the fact that we only needed the subjectivity and emotion dictionaries from the package. Furthermore, we chose to use the dictionaries accompanying the Sentiment package because we were unable to find any other resources that contained an emotions dictionary with a subjectivity dictionary. Also, in order to conform to the current literature on sentiment analysis, we chose to score the subjectivity of each politician by analyzing each tweet separately, but we chose to score the emotion of each politician by analyzing the whole corpus for that politician.

# Results

## Classification using K-Nearest Neighbours

Using KNN to identify the politicians of interest for this report yielded an accuracy of around 66%. Due the inherent noise within Twitter data, this result met expectations. Furthermore, given that the politicians were all discussing similar issues while trying to make themselves sound as appealing as possible so as to draw as many supporters as possible, an accuracy of 66% is within reason. The confusion matrix in figure 5 outlines the result of the classification:

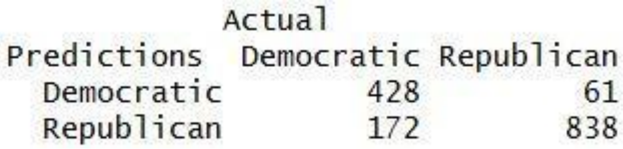


```
             Actual
Predictions Bush Clinton Cruz Sanders Trump
    Bush     181      53   44      24    16
    Clinton   18     172   14      30     7
    Cruz      78      61  254      31    49
    Sanders   14      15   10     196    18
    Trump      8       5    0       4   197
```

*Figure 5* - KNN Classification Results for Identifying Politicians

As can be seen, one interesting result is that not many politicians were being misclassified as Donald Trump, but Donald Trump was sometimes being misclassified as other politicians. Also, it seems that Bernie Sanders and Donald Trump were more easily correctly classified than Jeb Bush, Hillary Clinton or Ted Cruz. This may mean that Sanders and Trump are more unique in their opinions and thus easier to classify than Bush, Clinton or Cruz despite belonging to different political parties.

This brings us to the accuracy of classifying politicians by political affiliation. The KNN classification performed well in this regard, yielding an accuracy of around 84%. Again, based on the inherent noise within Twitter data and the fact that the politicians are all discussing the same issues, an accuracy of 84% attests to the success of the KNN classification algorithm. The confusion matrix below in figure 6 outlines the result of the classification:



```
             Actual
Predictions  Democratic Republican
  Democratic        428         61
  Republican        172        838
```

*Figure 6* - KNN Classification Results for Separating Politicians by Political Party

As can be seen, the model was quite accurate in correctly classifying a Republican politician while performing less well in correctly classifying a Democratic politician. This may be due to Republicans only discussing a smaller subset of issues whereas Democrats seem to be more likely to discuss a wider variety of issues. Hence, Democrats might be more likely to touch on topics similar to the interest of Republican politicians while a Republican will not reciprocate.

In regards to highlighting the issues relevant to each politician, we were able to create “word clouds” based on the 1000 tweets for each politician (See Figures 14-18). These were a great way to identify the most commonly used words and phrases for each politician, as seen in figure 14 “thank” in the middle was used most frequently by Bush while “kick” in the top was not used as often.

## Sentiment Analysis

These analyses were performed using previously compiled dictionaries containing words and the associated subjectivity or emotion attributed to that word. We were interested in seeing whether certain politicians would be more negative in their subjectivity or would portray negative emotions more frequently. As it turns out, it seems all the politicians were fairly neutral in regards to the subjectivity of their tweets and seemed to most often portray the emotion of joy. Figures 7 and 8 below outline the subjectivity of the Democrat, Bernie Sanders and the Republican, Donald Trump:

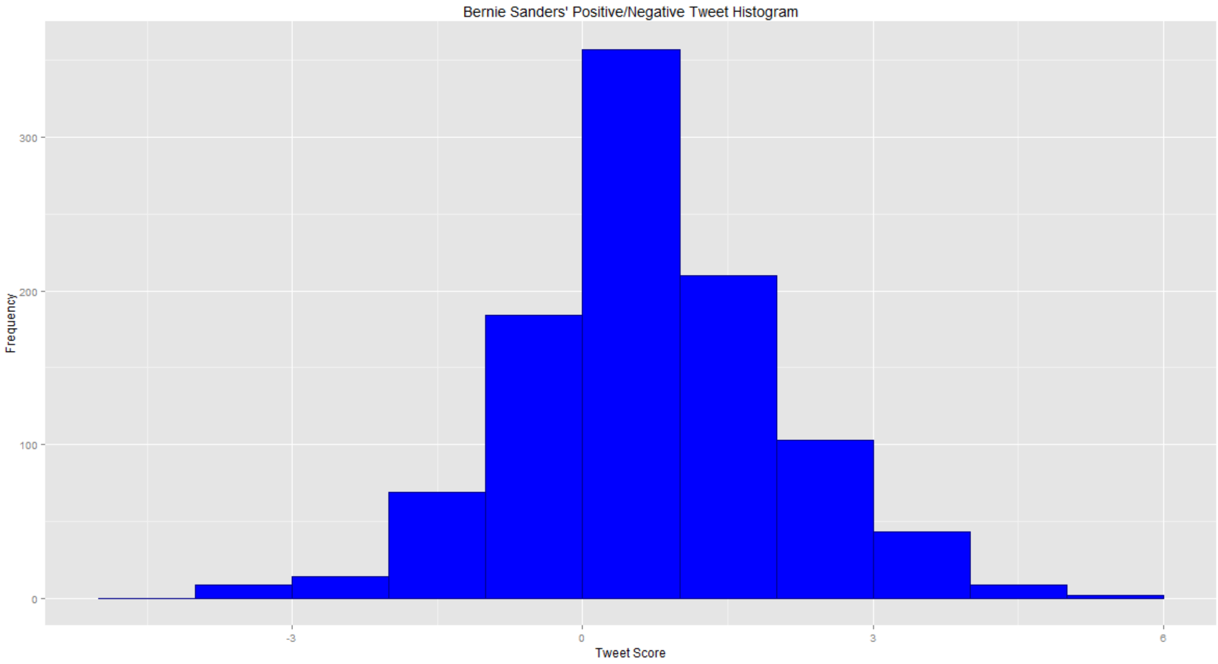


*Figure 7* - Bernie Sanders Tweet Subjectivity Distribution

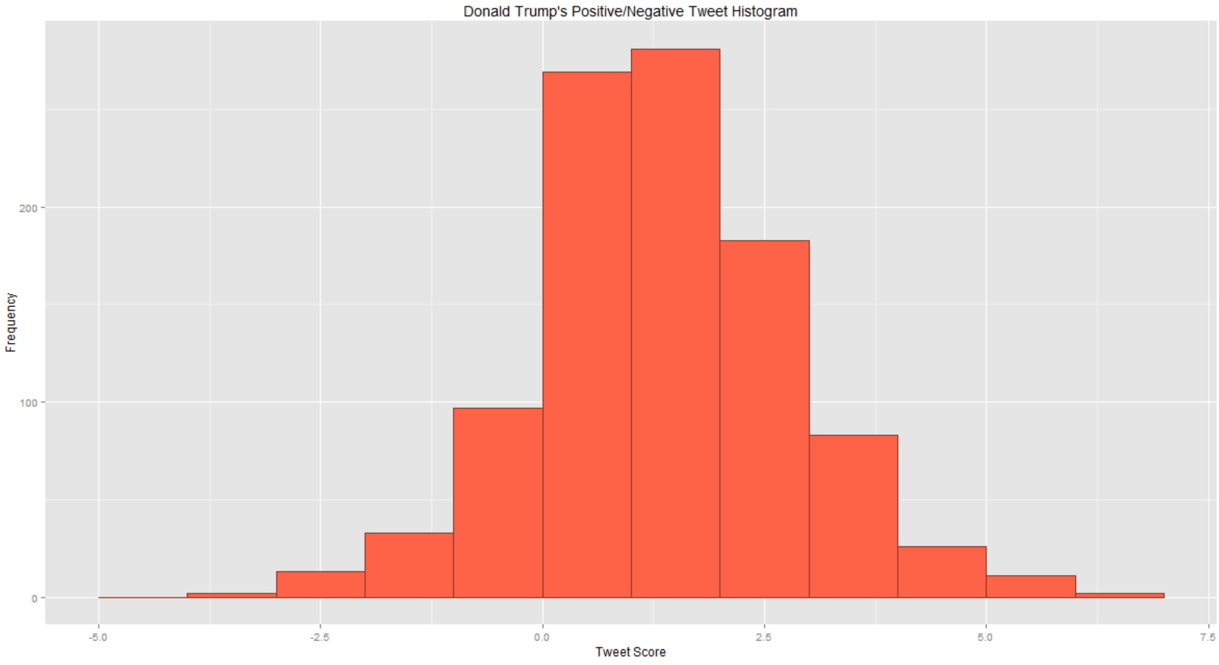
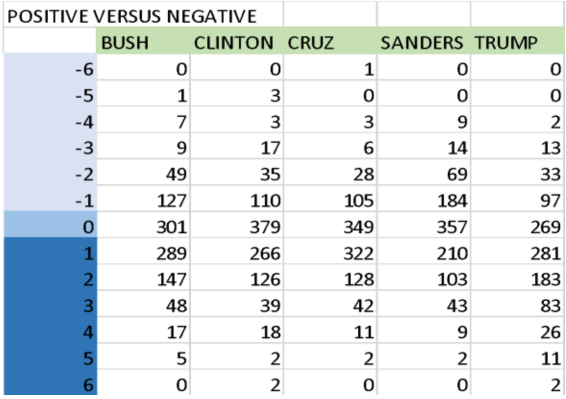


*Figure 8* - Donald Trump Tweet Subjectivity Distribution

As can be seen, even two arguably diametrically opposed politicians had similar subjectivity profiles. This may seem surprising at first, but once again it is important to note that the politicians all aim to appeal to as wide an audience as possible to ensure they garner maximum support for their cause.

Furthermore, the table in figure 9 succinctly outlines the subjectivity score breakdown associated with each politician. The more negative the value, the more negatively the tweet was scored. Once again, it is readily apparent that the politicians seemed to avoid any overly negative or overly positive expression in their tweets.



POSITIVE VERSUS NEGATIVE

| | BUSH | CLINTON | CRUZ | SANDERS | TRUMP |
|---|---|---|---|---|---|
| -6 | 0 | 0 | 1 | 0 | 0 |
| -5 | 1 | 3 | 0 | 0 | 0 |
| -4 | 7 | 3 | 3 | 9 | 2 |
| -3 | 9 | 17 | 6 | 14 | 13 |
| -2 | 49 | 35 | 28 | 69 | 33 |
| -1 | 127 | 110 | 105 | 184 | 97 |
| 0 | 301 | 379 | 349 | 357 | 269 |
| 1 | 289 | 266 | 322 | 210 | 281 |
| 2 | 147 | 126 | 128 | 103 | 183 |
| 3 | 48 | 39 | 42 | 43 | 83 |
| 4 | 17 | 18 | 11 | 9 | 26 |
| 5 | 5 | 2 | 2 | 2 | 11 |
| 6 | 0 | 2 | 0 | 0 | 2 |

*Figure 9* - Subjectivity Score for Each Politician

For reference, below in figure 10 are a few examples of tweets that were scored positive, negative or neutral based on the scoring functions created for this report:

**Trump:**
**Negative (-4 or worse):**
"I have watched sloppy Graydon Carter fail and close Spy Magazine and now am watching him fail at @VanityFair Magazine. He is a total loser!"
".@KarlRove is a biased dope who wrote falsely about me re China and TPP. This moron wasted $430 million on political campaigns and lost 100%"
**Neutral (0 score):**
"Ben Carson has never created a job in his life (well, maybe a nurse). I have created tens of thousands of jobs, it's what I do."
**Positive (+4 or better):**
"I had a great time in Texas yesterday. A tremendous crowd of wonderful and enthusiastic people. Will be back soon!"
"Saturday Night Live has some incredible things in store tonight. The great thing about playing myself is that it will be authentic! Enjoy"

*Figure 10* - Examples of Tweets and Accompanying Subjectivity Score

Similar figures for tweet subjectivity for the remaining politicians can be found in Appendix A (Figures 19-29).

When analyzing the dominant emotion in each politician's complete set of tweets it was easily apparent that they all portrayed their emotions in a "joyful" manner, largely avoiding emotions with negative connotations such as "disgust", "fear" or "anger." Figures 11 and 12 below outline the breakdown of the emotion for the tweets of Democrat, Hillary Clinton and Republican, Jeb Bush:

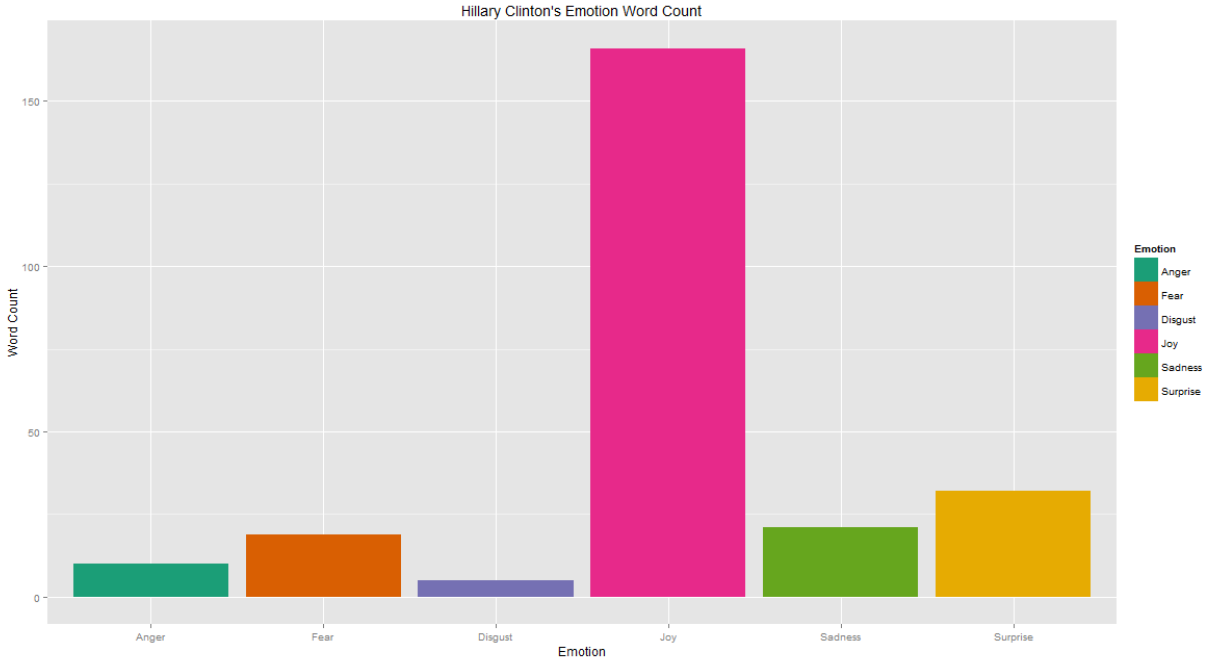


*Figure 11* - Hillary Clinton's Tweet Emotion Breakdown

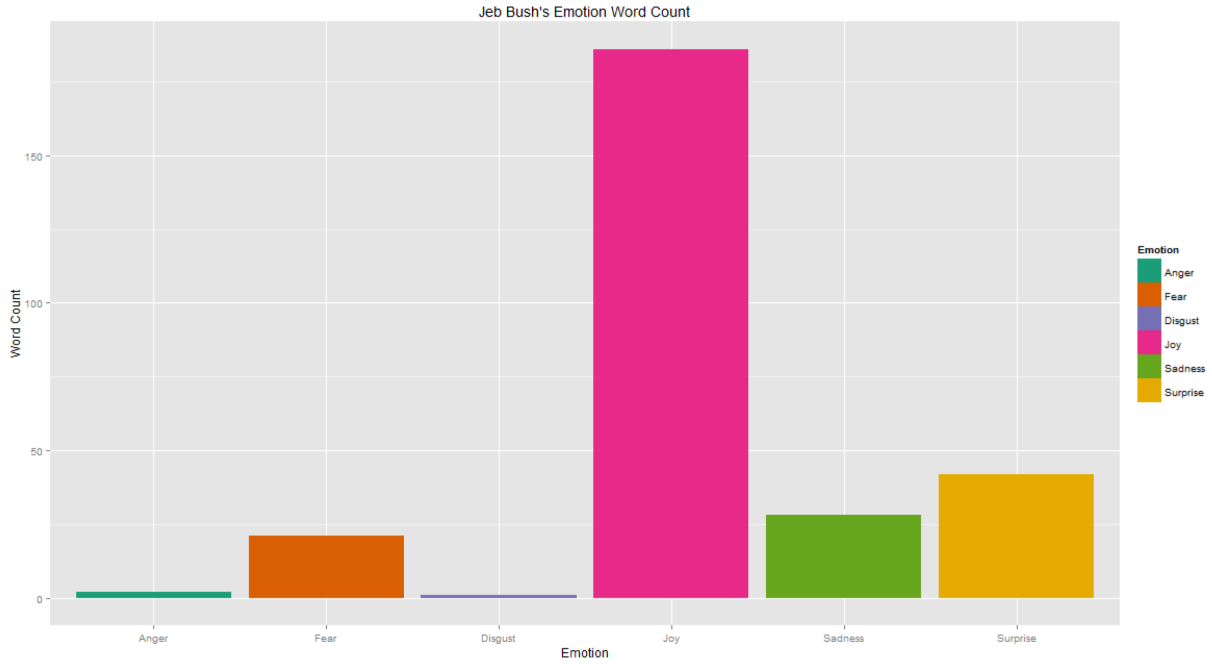
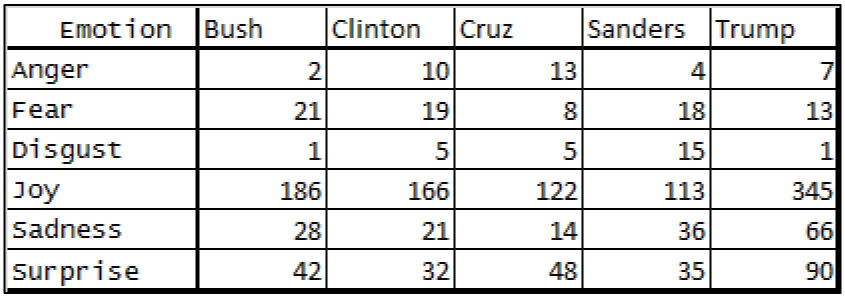


*Figure 12* - Jeb Bush's Tweet Emotion Breakdown

As can be seen, even though Clinton and Bush come from opposing political parties and have been active in discrediting one another's platforms and opinions, they both seem to present a similar emotional profile in their tweets. This result matches what we have already seen with regard to the subjectivity profile of each politician. It seems that the politicians are determined to present an optimistic undertone to their tweets, regardless of the issue they are currently discussing. This could arguably make the politician seem like a promising candidate to lead the country through issues and conflict that may arise in the future. Furthermore, the table in figure 13 succinctly outlines the emotion word count associated with each politician.



| Emotion | Bush | Clinton | Cruz | Sanders | Trump |
|---|---|---|---|---|---|
| Anger | 2 | 10 | 13 | 4 | 7 |
| Fear | 21 | 19 | 8 | 18 | 13 |
| Disgust | 1 | 5 | 5 | 15 | 1 |
| Joy | 186 | 166 | 122 | 113 | 345 |
| Sadness | 28 | 21 | 14 | 36 | 66 |
| Surprise | 42 | 32 | 48 | 35 | 90 |

*Figure 13* - Emotion Word Count for Each Politician

Here, it is readily apparent that the politicians seemed to most often project the emotion of "joy" while avoiding overly negative emotions such as "anger", "fear" or "disgust". Trump's tweets convey a lot of words of joy, however, this large proportion is made greater due to his campaign slogan of "Make America Great Again" which is repeated in numerous tweets. Our dictionaries classify "great" as a joyous word, and thus, Trump appears excessively joyous due to his slogan. Once again, by coming across as positive individuals, the politicians may be hoping to market themselves as an optimistic and appealing potential leader. Similar figures for emotion word count for the remaining politicians can be found in Appendix A (Figures 30-34).

## Discussion

KNN was used to classify/identify politicians and party affiliation by tweets. The results of tweet classification by politician may be found in the confusion matrix in figure 5. The results of tweet classification by political party may be found in the confusion matrix in figure 6. The accuracy for the classification of politician by tweet was 66% while the accuracy for classification by party was 84%. Given the high degree of noise from Twitter data and the similarity in subject matter tweeted by each politician, these results met expectations. Some further reasons for the lack of accuracy could be due to the algorithm used, KNN. The choice of sparsity tuning factor may have played a role in the accuracy. The processing of the data may have affected the results as well. The sentiment analysis revealed that all candidates were more positive or neutral than negative. This can be seen in the tweet subjectivity distribution graphs in figures 19-29 (Appendix A) for all the candidates considered. The breakdown of emotions for each candidate also reveals a common trend. As seen in figures 30-34 (Appendix A), all candidates show a great deal of joy through their tweets. This positive, joyful sentiment expressed by each candidate is not a complete surprise. The context of all the tweets is relatively expressing the same theme, that of directness through carefully chosen words for the purpose of positively branding themselves to the public. Possible reason for the lack of accuracy could be due to the score function used; the method used for the sentiment analysis; the dictionaries used and the words in them. Not all tweeted words may have been found in the chosen dictionaries. Some possible areas of further research could include using a corpus that included tweets from different days and/or times during the week. The classification by candidate and party could be done using another classification method and the results and accuracy compared to those found in this study. While sentiment analysis was performed on the entire set of 1000 tweets for each candidate going one step further and determining the most representative topic behind the sentiment through topic recognition would be an interesting extension.

# References

1. Emotion space model for classifying opinions in stock message board, Banghui Luo, Jianping Zeng, Jiangjiao Duan, Expert Systems with Applications, 2016, Vol. 44, p. 138-146
2. Classifying Sentiment in Microblogs: Is Brevity an Advantage?, Adam Bermingham, Alan Smeaton
3. Text mining and topic models, Charles Elkan, February 2014.
4. Introduction to the tm Package Text Mining in R, Ingo Feinerer, July 2015.
5. The Text Mining Handbook, Advanced Approaches in Analyzing Unstructured Data, Ronen Feldman, James Sanger, Cambridge University Press, 2007.
6. Twitter brand sentiment analysis: A hybrid system using n-gram and dynamic artificial neural network, M. Ghiassi, J. Skinner, D. Zimbra, Expert Systems with Applications: An International Journal, 2013, Vol. 40, Iss. 16, p. 6266-6282.
7. A Survey of Text Mining Techniques and Applications – Vishal Gupta, Gurpreet S. Lehal, Journal of Emerging technologies in Web Intelligence, Vol !., No 1, August 2009
8. Social media competitive analysis and text mining: A case study in the pizza industry, Wu He, Shenghua Zha, Ling Li, Volume 33, p. 464-472, 2013.
9. A Brief Survey of Text Mining – Andreas Hotho, Andreas Nurnberger, Gerhard Paab, May 2005
10. Opinion Mining, Sentiment Analysis, and Opinion Spam Detection (Hu and Liu's lexicon), URL: https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html
11. R - Twitter Mining with R (part 1), Jalayer Academy URL: https://www.youtube.com/watch?v=lT4Kosc_ers
12. Text-mining and information-retrieval service for molecular biology, Martin Krallinger and Alfonso Valencia, Genome Biology, Volume 6, Iss. 7, Article 224, 2005.
13. Practical text mining and statistical analysis for non-structured text data applications, Miner, Gary
    Elder, John, IV, Hill, Thomas, Academic Press, 2012.
14. Social Media Analytics: Data Mining Applied to Insurance Twitter Posts, Roosevelt C. Mosley Jr., Casualty Actuarial Society E-Forum, Winter 2012, Volume 2, p. 1-36.
15. More than words: Social networks' text mining for consumer brand sentiments, Mohamed M. Mostafa, Expert Systems with Applications Volume 40, p. 4241 – 4251, 2013.
16. Effective Pattern Discovery for Text Mining, Ning Zhong, Yuefeng Li, Sheng-Tang Wu, Knowledge and Data Engineering, IEEE Transactions, 2012, Vol. 24, Iss. 1, p. 33-44.
17. Research Challenge in Opinion Mining and Sentiment Analysis, David Osimo, Francesco Mureddu
18. R gsub Function, URL: http://www.endmemo.com/program/R/gsub.php
19. Sentiment analyses, URL : https://en.wikipediva.org/wiki/Sentiment_analysis
20. SentiWordNet, URL: http://sentiwordnet.isti.cnr.it/
21. What is the difference between emotion recognition and sentiment analysis? URL: https://www.quora.com/What-is-the-difference-between-emotion-recognition-and-sentiment-analysis
22. Relevance Feature Discovery for Text Mining, Yuefeng Li, A. Algarni, M. Albathan, Yan Shen, Knowledge and Data Engineering, IEEE Transactions, 2015, Vol. 27, Iss. 6, p. 1656-1669.
23. Text Mining with R – Twitter Data Analysis, Yanchang Zhao, URL: http://www.rdatamining.com/docs/text-mining-with-r-of-twitter-data-analysis
24. Basic Text Mining in R, URL: https://rstudio-pubs-static.s3.amazonaws.com/31867_8236987cf0a8444e962ccd2aec46d9c3.html
25. Text Mining of Presidential Campaign Speeches in R - Romney vs. Obama, Timothy DAuria, URL: https://www.youtube.com/watch?v=2Znairz-hvU

26. How to Build a Text Mining, Machine Learning Document Classification System in R!, Timothy DAuria , URL: https://www.youtube.com/watch?v=j1V2McKbkLo
27. Text Mining in R Tutorial: Term Frequency & Word Clouds, deltaDNA, URL: https://www.youtube.com/watch?v=lRTerj8fdY0
28. Remove URLs from text, Yanchang Zhao, URL: http://www.rdatamining.com/books/rdm/faq/removeurlsfromtext
29. Hierarchal Clustering, URL: https://rstudio-pubs-static.s3.amazonaws.com/31867_8236987cf0a8444e962ccd2aec46d9c3.html#hierarchal-clustering
30. Package "twitteR", Gentry J., July 29, 2015. https://cran.rproject.org/web/packages/twitteR/twitteR.pdf
31. Introduction to the tm Package Text Mining in R, Feinerer I., July 3, 2015. https://cran.r-project.org/web/packages/tm/vignettes/tm.pdf
32. Package 'class', Ripley B., Venables W., August 30, 2015. https://cran.rproject.org/web/packages/class/class.pdf
33. RCV1: A New Benchmark Collection for Text Categorization Research. Lewis D. R., Yang Y., Rose T., Li F., Journal of Machine Learning Research 5 (2004) 361-397.
34. Building subjectivity scoring function: https://github.com/mjhea0/twitter-sentiment-analysis/blob/master/R/sentiment.R
35. Obtaining subjectivity and emotion dictionaries: https://cran.rproject.org/src/contrib/Archive/sentiment/

## Appendix A - Tables and Figures

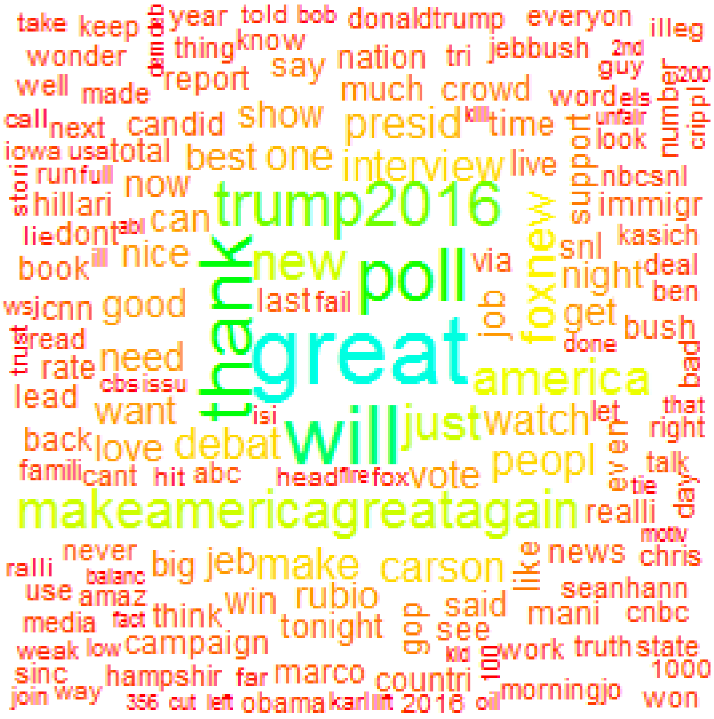


*Figure 14 - Trump Word Cloud*

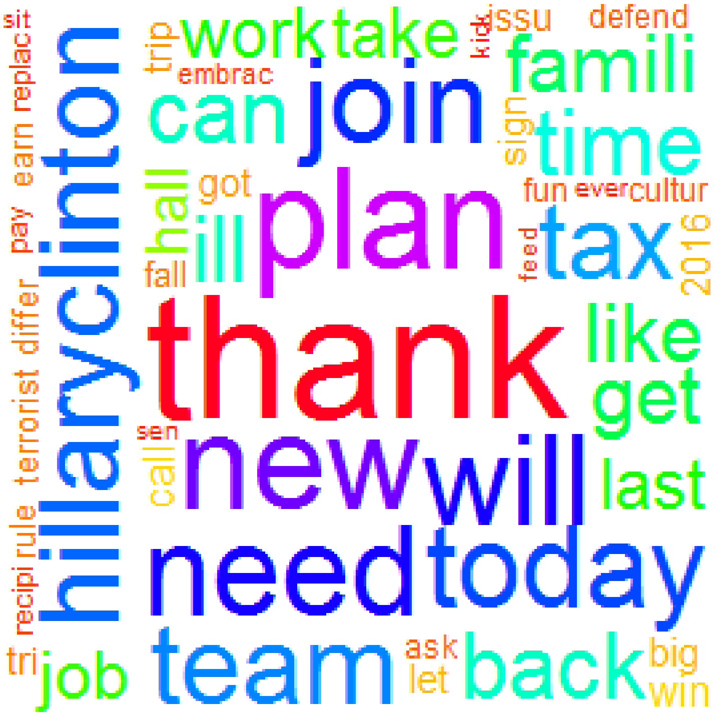


*Figure 15 - Bush Word Cloud*

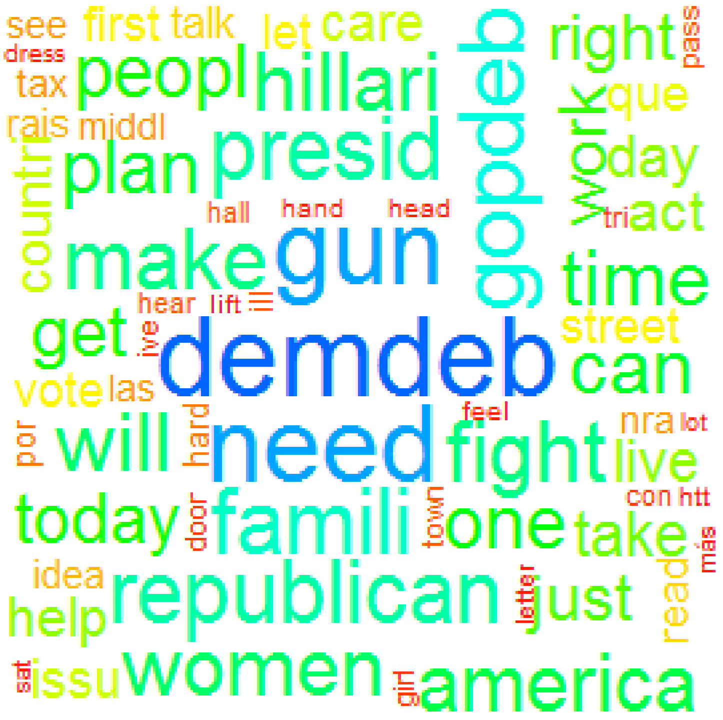


*Figure 16 - Clinton Word Cloud*

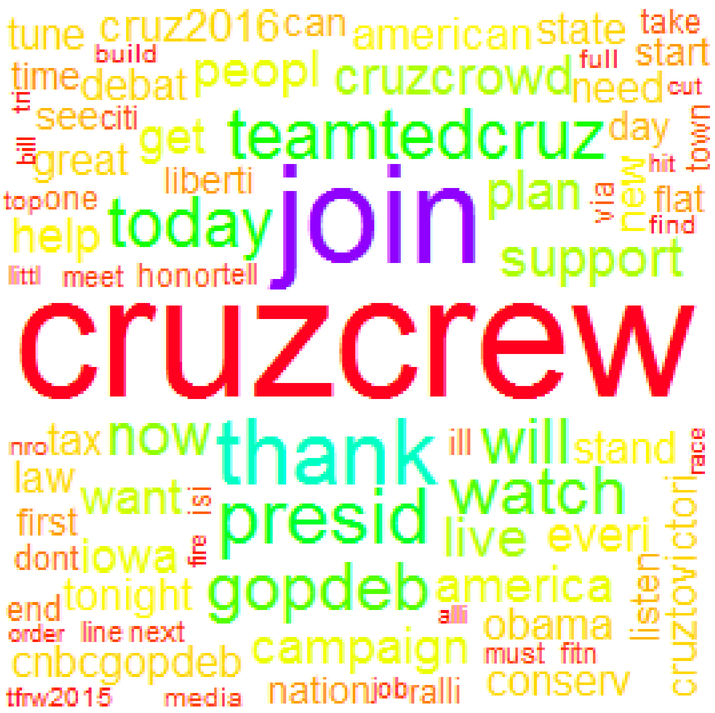


*Figure 17 - Cruz Word Cloud*

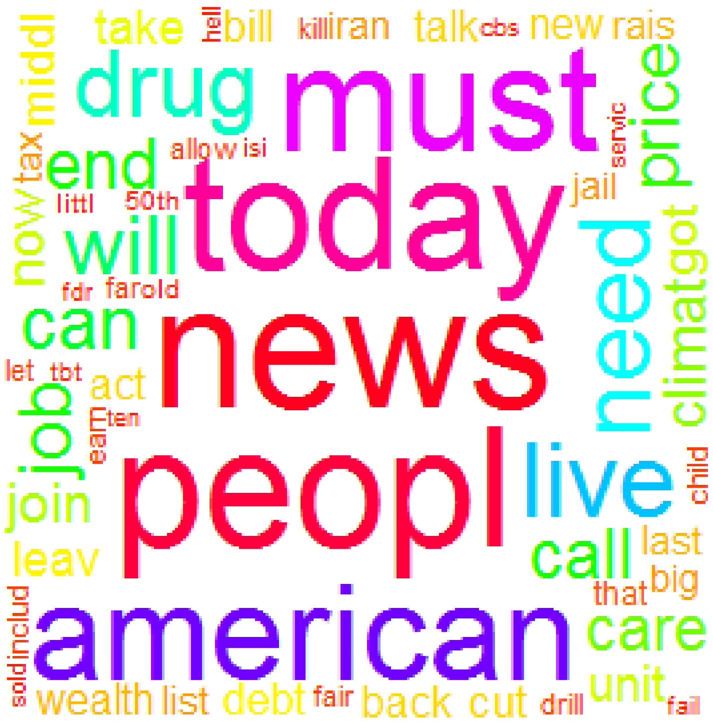


*Figure 18 - Sanders World Cloud*

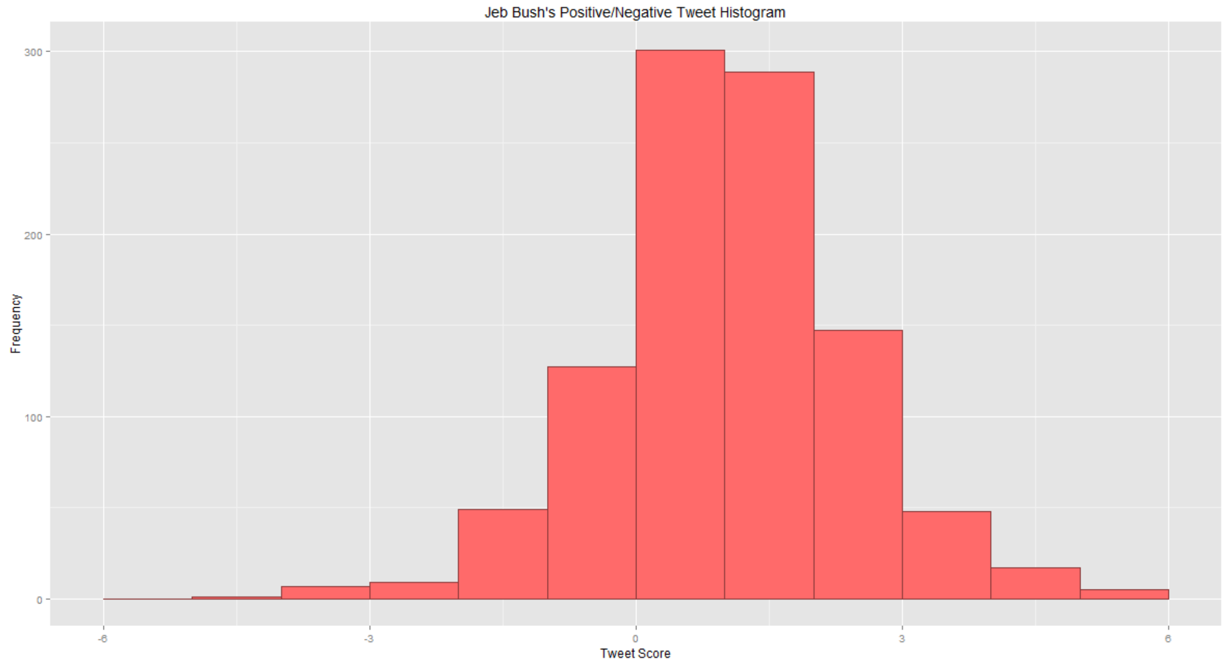


*Figure 19*- Bush Subjectivity

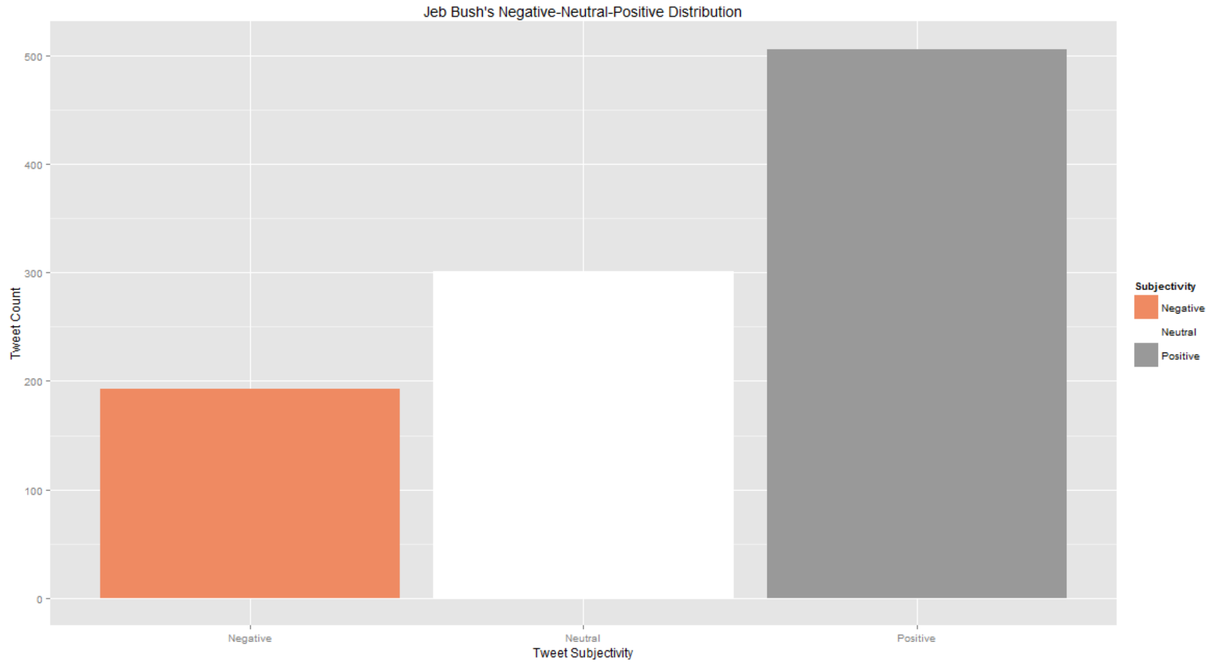


*Figure 20- Bush Subjectivity*

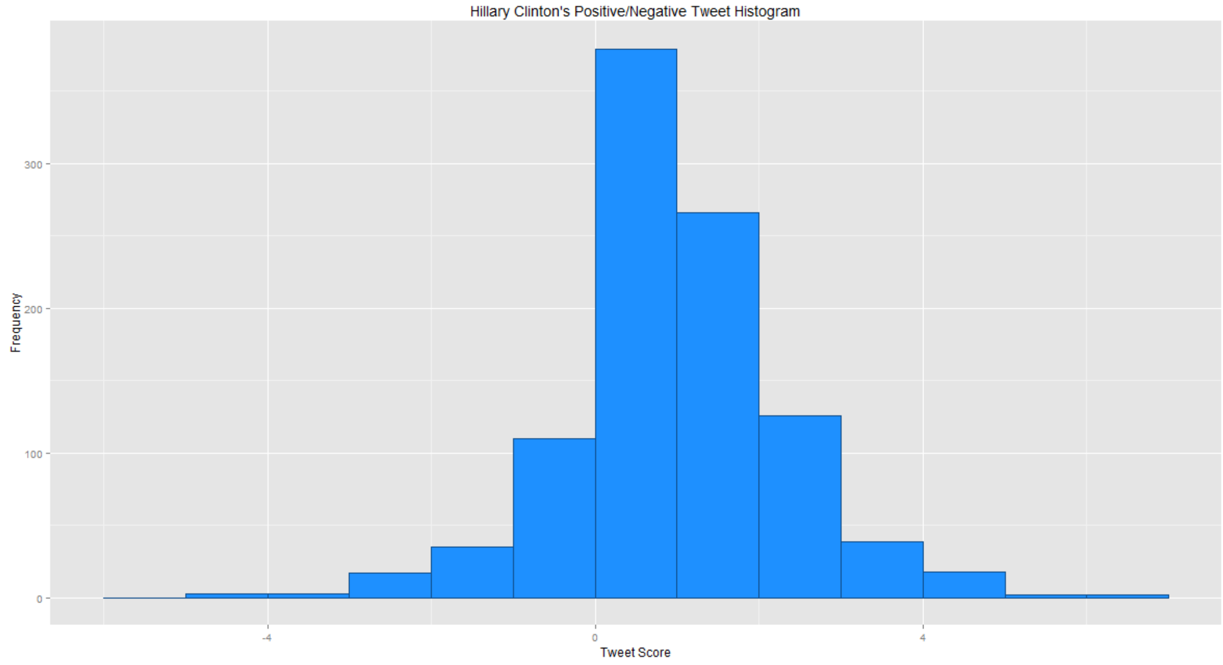


*Figure 21 - Clinton Subjectivity*

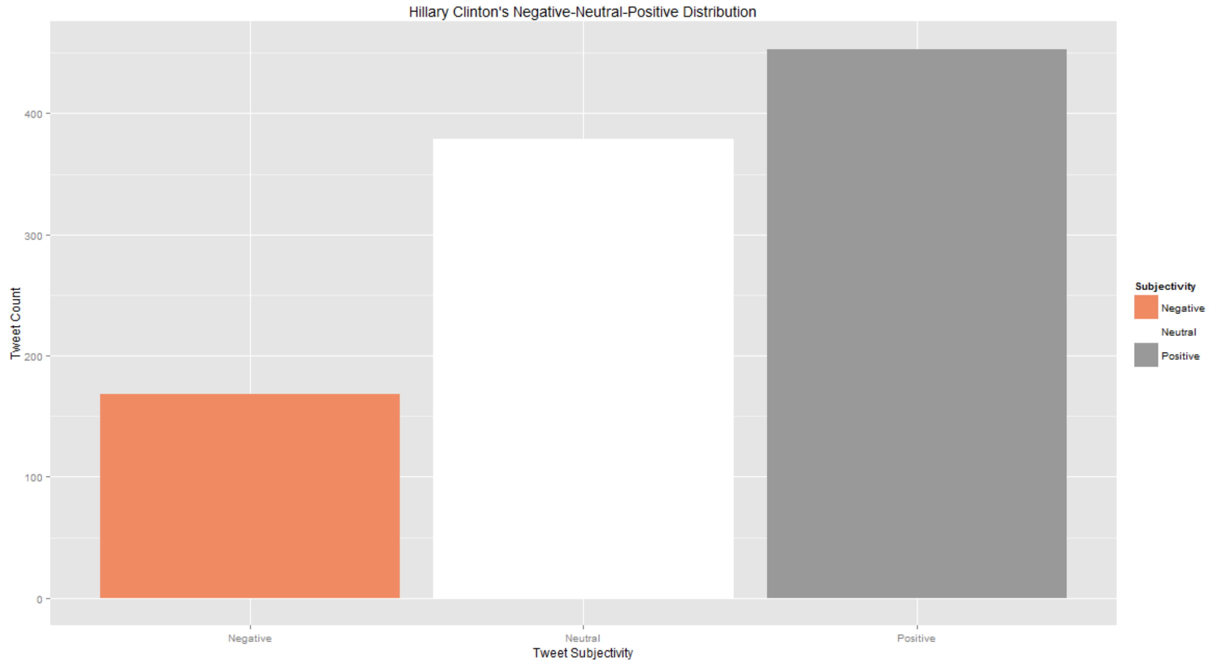


*Figure 22 - Clinton Subjectivity*

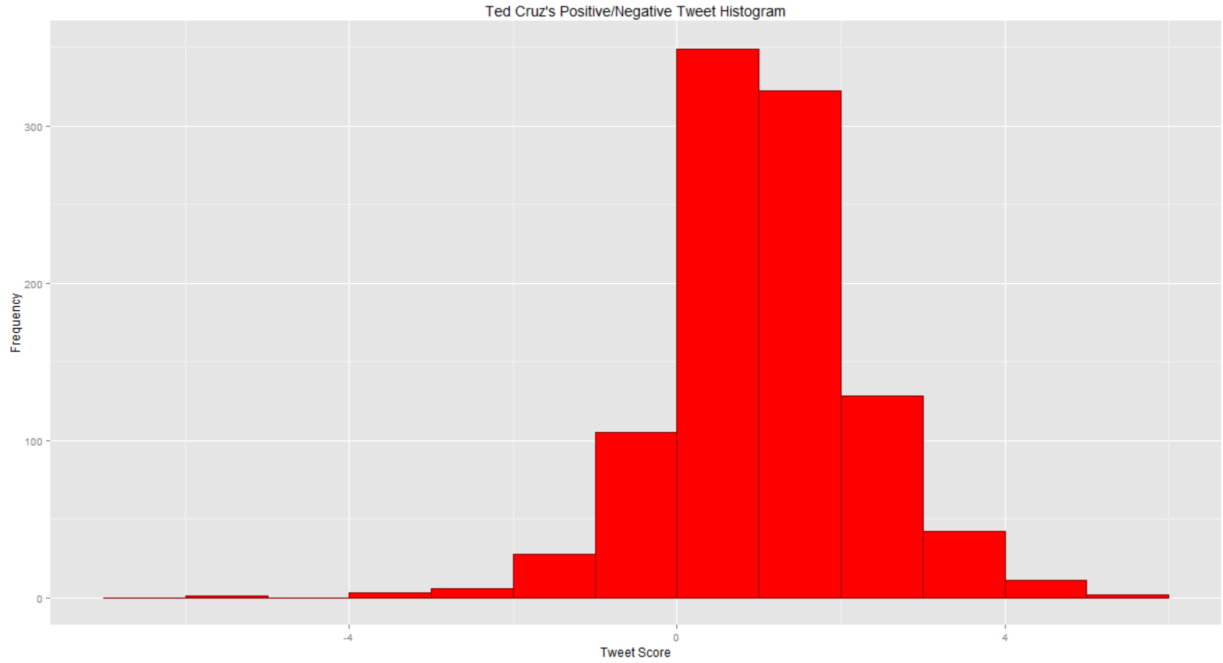


*Figure 23 - Cruz Subjectivity*

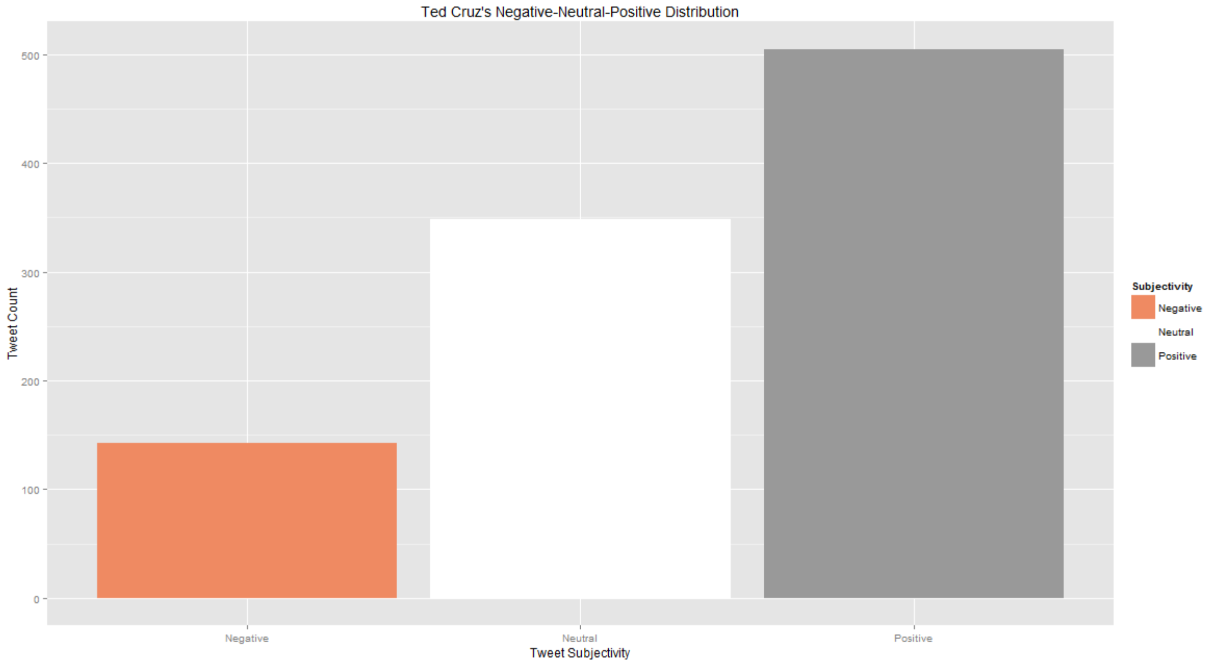


*Figure 24 - Cruz Subjectivity*

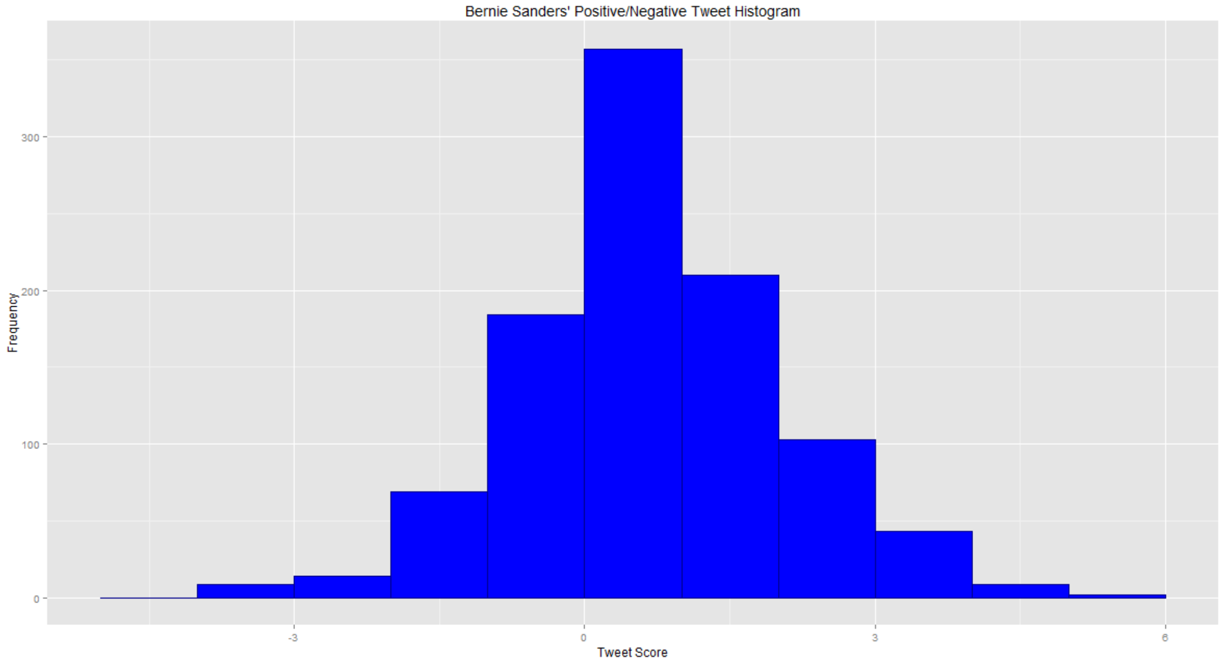


*Figure 25 - Sanders Subjectivity*

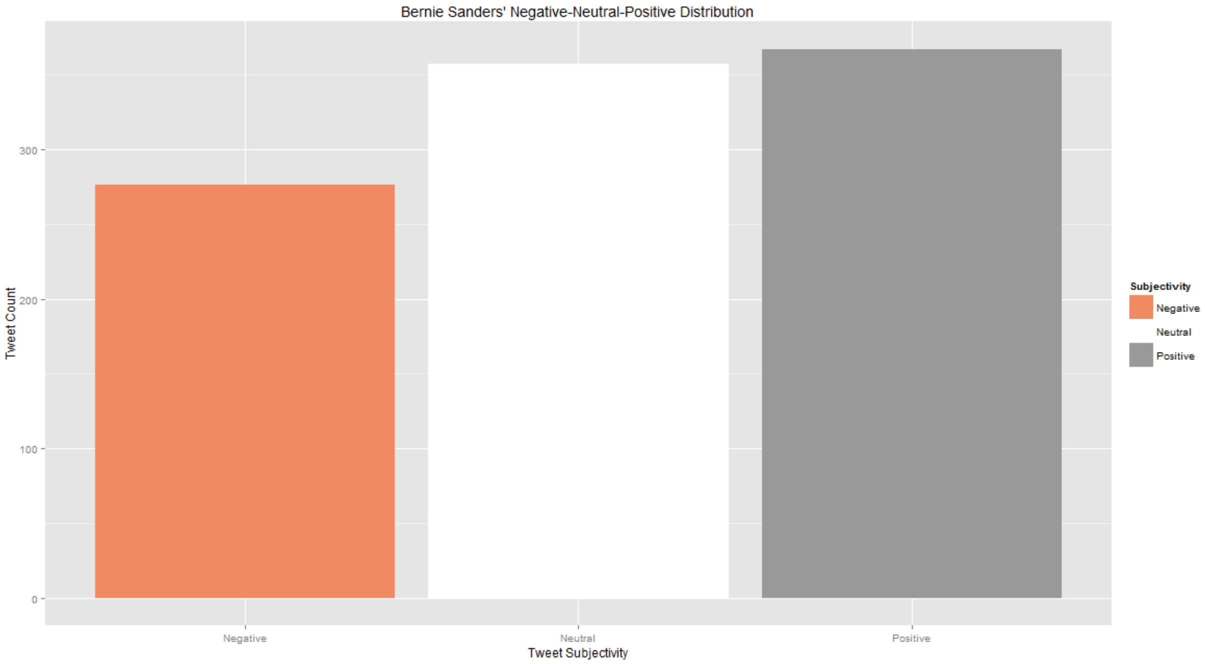


*Figure 26 - Sanders Subjectivity*

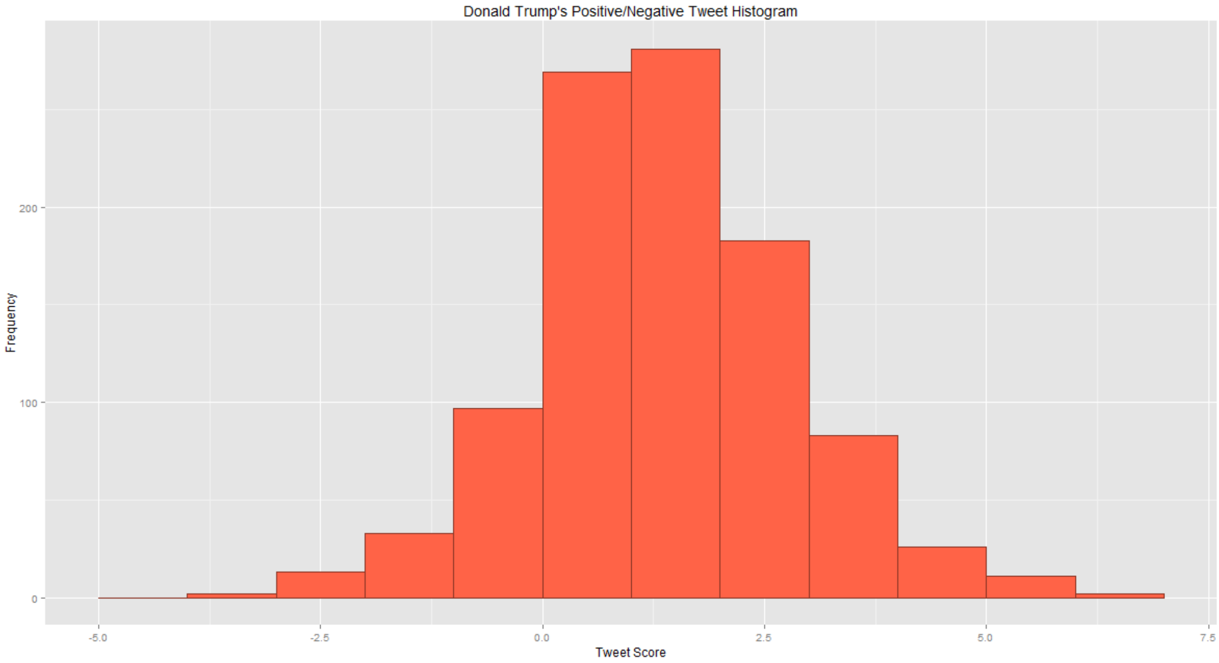


*Figure 27 - Trump Subjectivity*

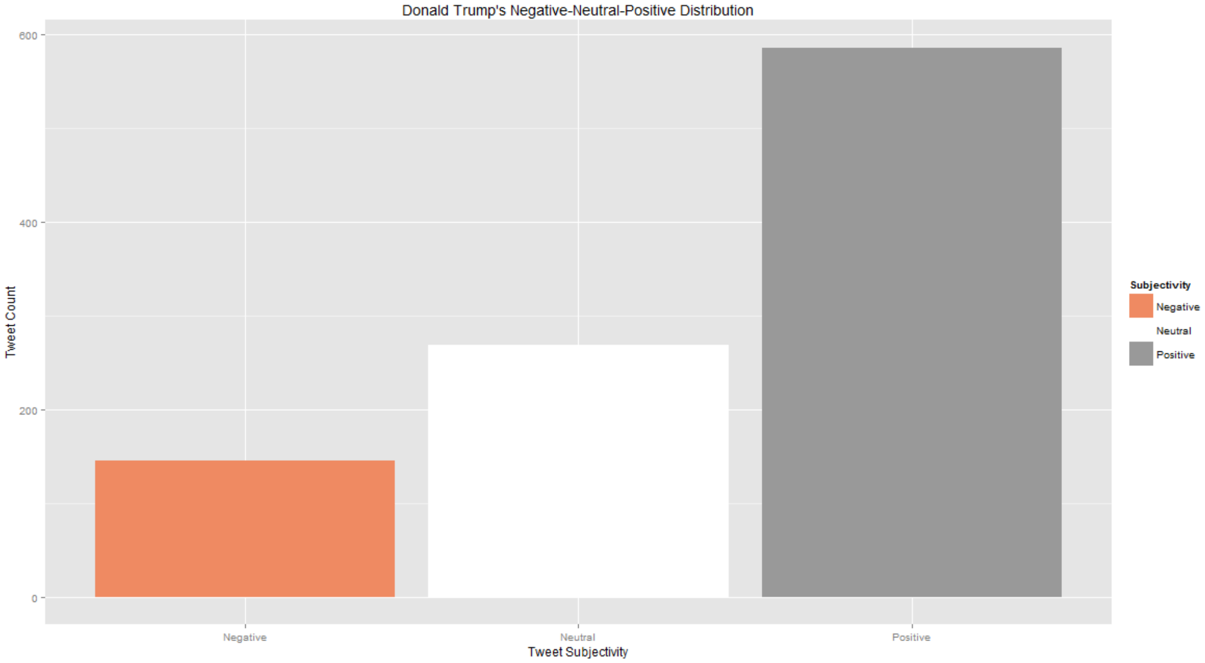
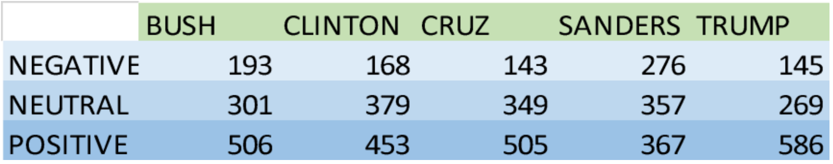


*Figure 28 - Trump Subjectivity*



|  | BUSH | CLINTON | CRUZ | SANDERS | TRUMP |
|---|---|---|---|---|---|
| NEGATIVE | 193 | 168 | 143 | 276 | 145 |
| NEUTRAL | 301 | 379 | 349 | 357 | 269 |
| POSITIVE | 506 | 453 | 505 | 367 | 586 |

*Figure 29 - All Politician's Subjectivity*

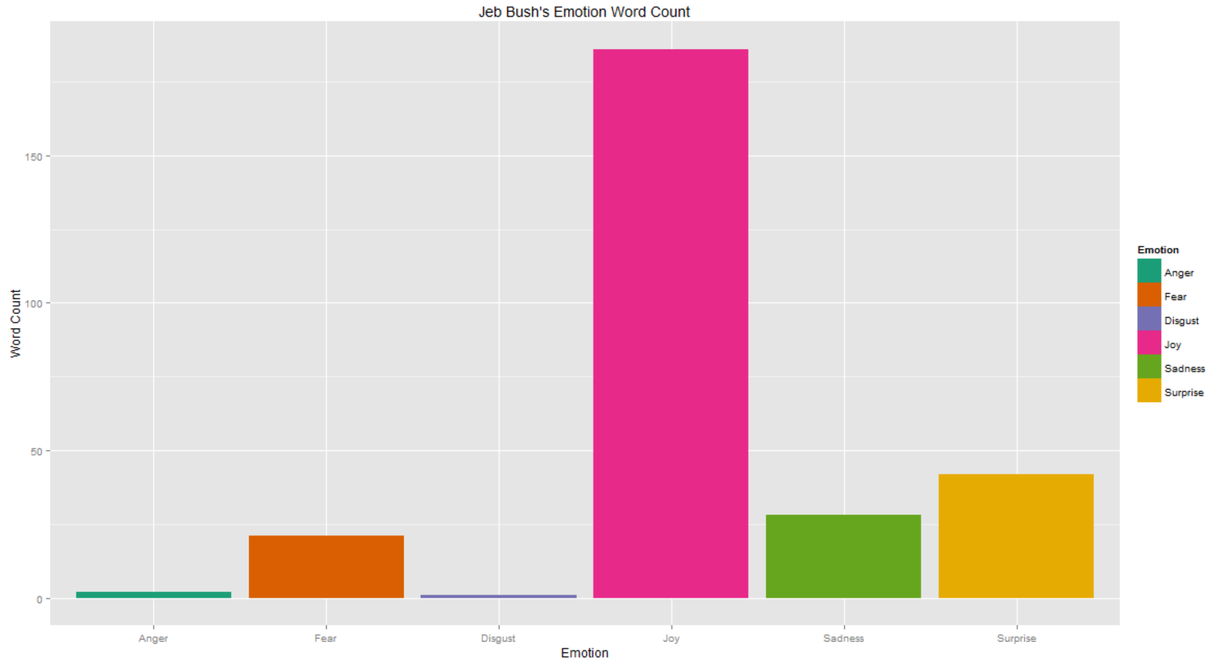


*Figure 30 - Bush Emotion*

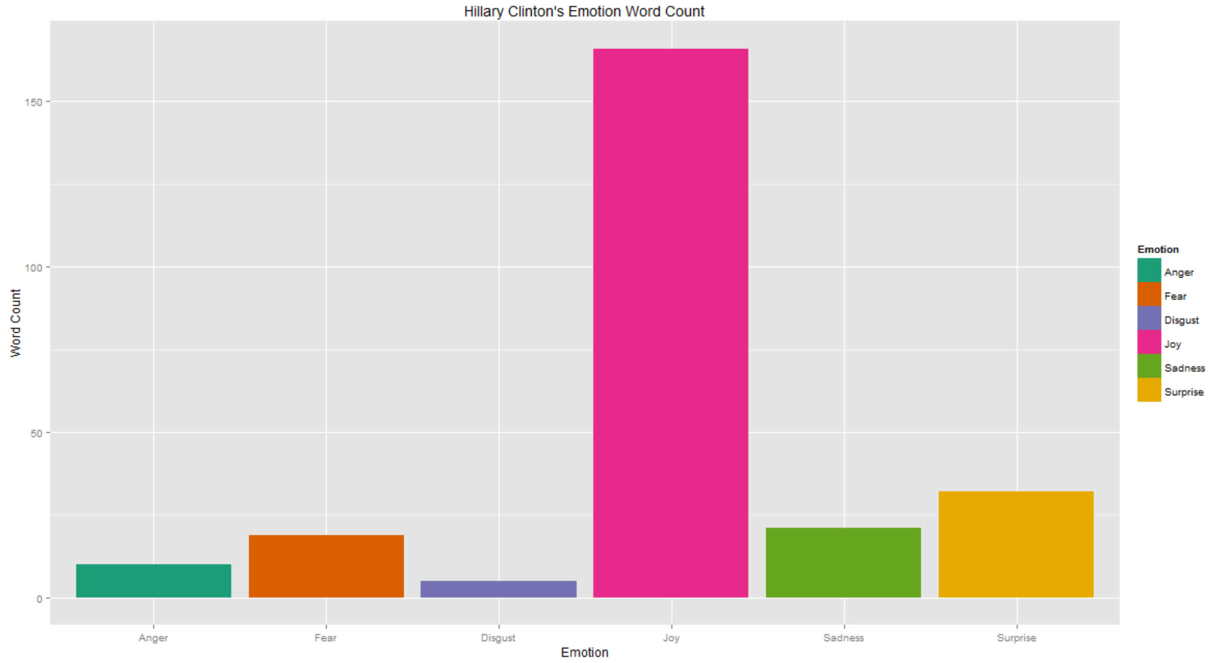


*Figure 31 - Clinton Emotion*

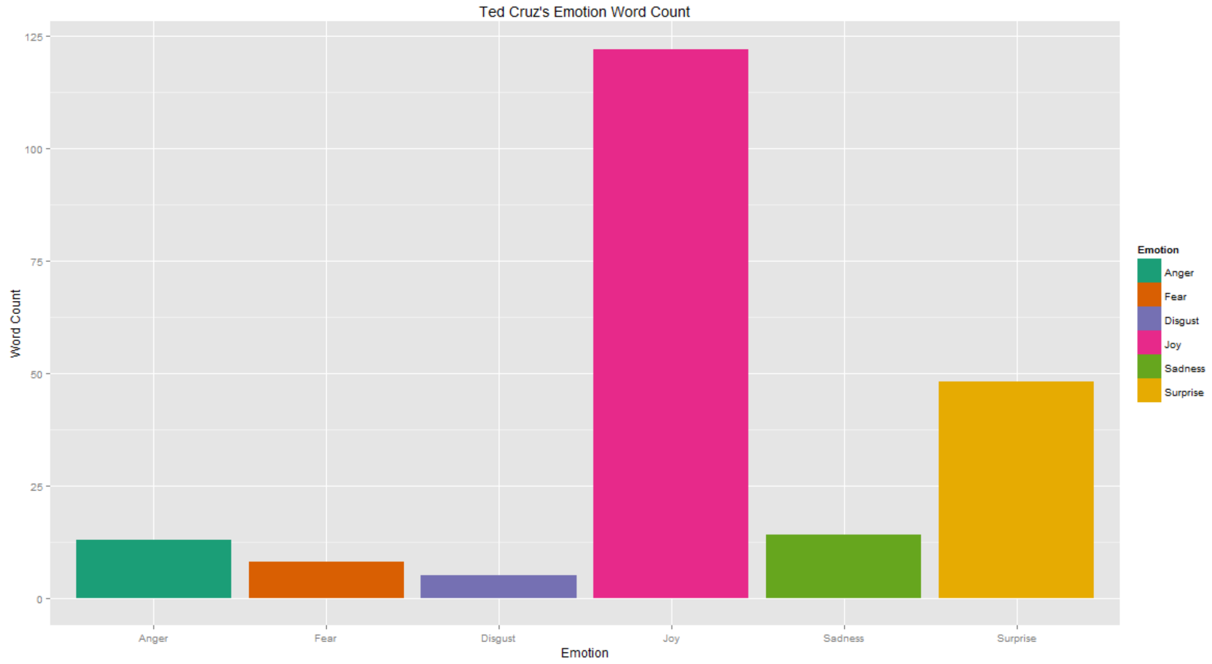


*Figure 32 - Cruz Emotion*

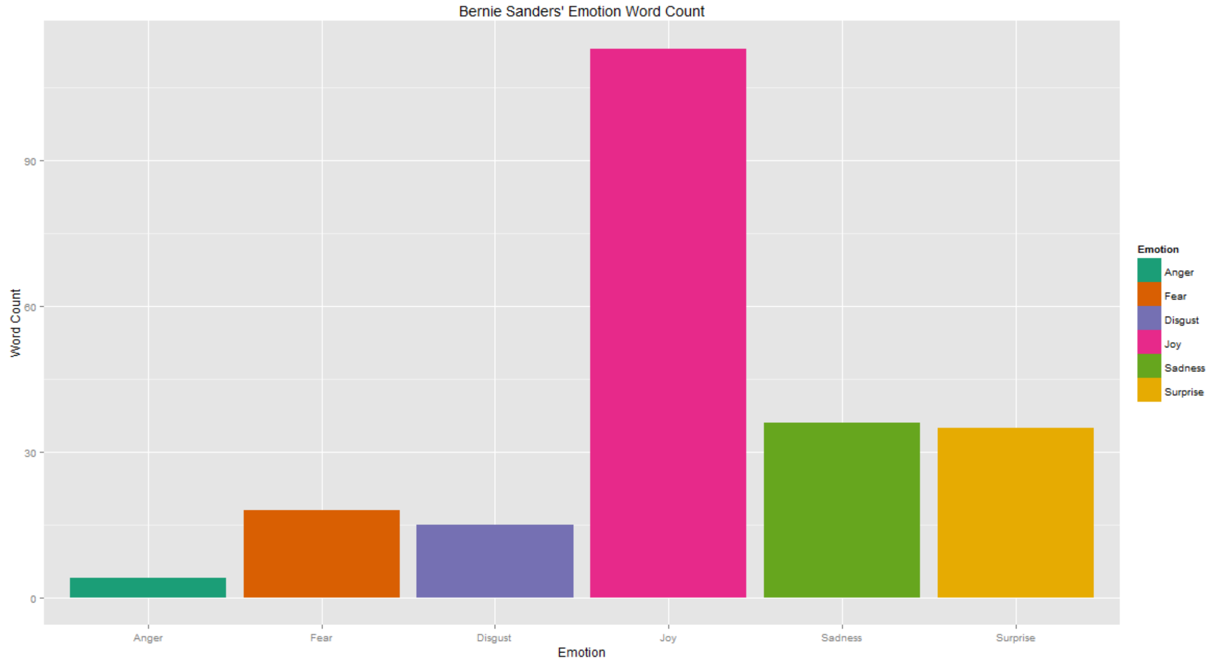


*Figure 33 - Sanders Emotion*

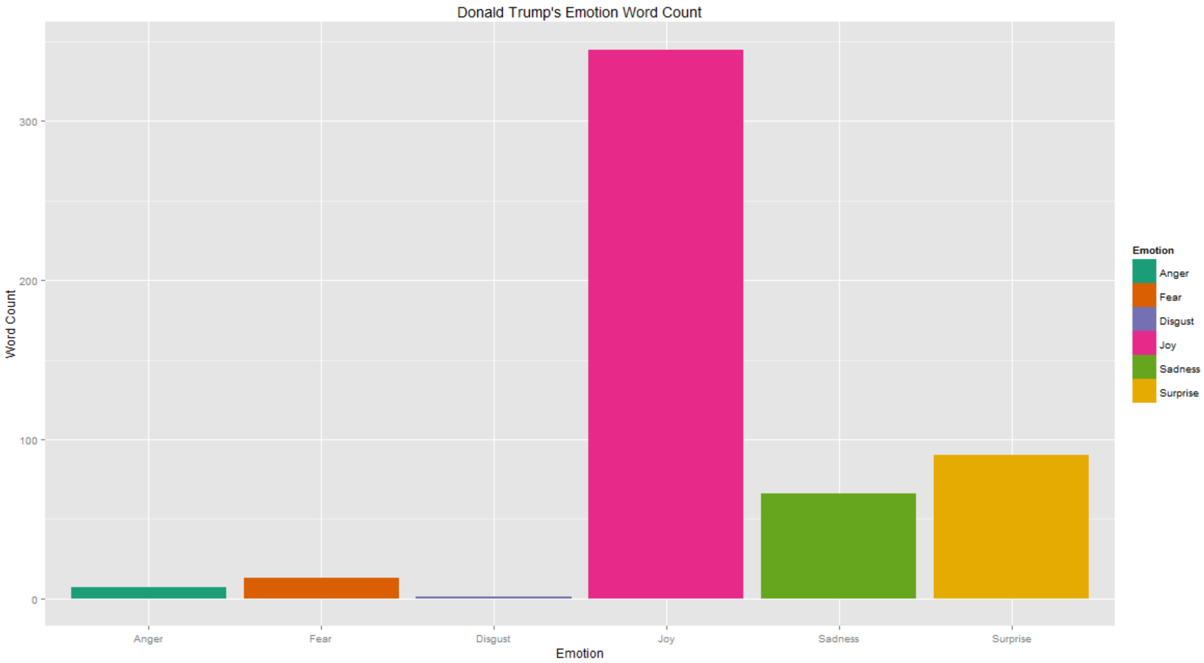


*Figure 34 -Trump Emotion*

## **Appendix B -** R-Code

Please see attached file **TEXT-MINING-ResearchProject-Code.txt**