Sentiment Analysis of Trump and Clinton’s Election Tweets

Ankamma Reddy Kolli

15/06/2022

## Abstract

Twitter is one of the most popular micro-blogging social media networking tools. Through this platform, there were many tweets related to the 2016 US Presidential election where people on their Twitter handles could express their views on a given presidential candidate. Therefore, it is better to understand the different sentiments associated with these tweets as presidential candidates expressed their views while doing their campaigns. The project was completed using a dataset downloaded from the Kaggle.com website. Using this dataset, Wordclouds for both candidates was added for this project. Besides this basic Wordcloud, a comparison cloud was added to the project to plot the difference in the word usage by these two presidential candidates. The project also calculates the sentiments expressed and then compares the identified sentiments and categorizes them as either positive or negative sentiments as used by the two candidates in the 2016 US Presidential election. The project utilized the lexicon algorithm that helped calculate the sentiment of political tweets collected 100 days before the election. The project utilized the manually labeled and automatically labeled tweets by looking into hashtag topics. The report suggests that Twitter is one of the most reliable platforms that are essential in providing previous work comparisons. From the project, it was possible to categorize the tweets and identify either of the two candidates who utilized the positive or negative sentiments in their campaigning tweets. From the results of this project, it would be recommended to perform sentiment analysis on the views based on people and their preferred presidential candidate and compare it with the polling data after the election.

## Introduction

Twitter is a social media networking platform that allows users to post tweets on their Twitter pages (Knight & Kaye, 2016). The tweets can be publicly viewed by anyone or made private only to be viewed by individuals who follow the user who posts the tweets. Twitter utilizes hashtags within tweets to identify different keywords or topics (Calvin et al., 2015). Therefore, this project aims to gain insights into different sentiments identified from tweets made by the two presidential candidates of the 2016 US Presidential elections, Donald Trump, the Republican candidate, and Hillary Clinton, the Democrat Candidate. The tweets could either be positive or negative, which could identify the candidate’s way of the campaign and the words. The project collects tweets from the two candidates from their Twitter pages. The project will map the tweets along with their sentiments from them. Therefore, the project creates a sentiment algorithm that will use a collection of words, each with its sentiment value, to analyze tweets for both Trump and Clinton 2016 US Presidential candidates.

## Literature Review

In a study by O’Connor et al. (2010), the authors gathered tweets from the 2012 US Presidential Election that was made up of the phrases Obama and McCain. In their study, they applied the lexicon sentiment analysis approach to the election tweets, from which they could compare the results with the polling data (O’Connor et al., 2010). A subjectivity lexicon was utilized, which was made up of approximately 1600 positive words and 1200 negative words. In another study by Jahnbakhsh & Moon, Naïve Bayes Algorithm was utilized to classify tweets associated with the 2012 US Presidential Election. Nine hundred eighty-nine tweets were manually labeled in this study to train the classifier from which the polling data and results were compared (Joyce & Deng, 2017). Wang et al. (2012), in their study, automatically labeled tweets based on emotions. The authors utilized hashtags labels such as #sorrow and #happy in training the classifier to identify tweets expressing sadness and joy. The authors were able to achieve a 65.65% accuracy in their analysis (Wang et al., 2012). From the related literature, the authors majorly concentrated on the comparison of sentiment results and polling data. However, since the presidential candidates have an influence on elections during their campaigns depending on what they say, it is necessary for this project where sentiment analysis will be performed on individual tweets to identify whether they contain positive or negative sentiments. The project intends to answer whether it is possible to categorize sentiment tweets as either positive or negative sentiments for the Democrat and Republican candidates.

## Theory

H1: The tweets from the Democrat candidate, Hillary Clinton, involve positive sentiments and emotions, but the tweets from the Republican candidate, Donald Trump, involve more negative sentiments and emotions and use more negation words.

## Data

Trump and Clinton's tweets are available at Kaggle: <https://www.kaggle.com/code/erikbruin/text-mining-the-clinton-and-trump-election-tweets/data>

The dataset is made up of 28 variables and 6,444 tweets. The variable handle is the dataset containing tweets from Donald Trump and Hillary Clinton. Within the dataset, if there is a blank in the “original\_author,” then either Hillary Clinton or Donald Trump is the handler. However, if no blank is left on the “original\_author” variable, it means that Clinton or Trump retweeted. For the text in these records, they are the original tweets and not retweets from Clinton or Trump.

tweets$time <- ymd\_hms(tweets$time)

glimpse(tweets)

## Rows: 6,444

## Columns: 28

## $ id <int64> 780925634159796224, 780916180899037184, 7809…

## $ handle <chr> "HillaryClinton", "HillaryClinton", "HillaryCl…

## $ text <chr> "The question in this election: Who can put th…

## $ is\_retweet <lgl> FALSE, TRUE, TRUE, FALSE, FALSE, FALSE, FALSE,…

## $ original\_author <chr> "", "timkaine", "POTUS", "", "", "", "", "", "…

## $ time <dttm> 2016-09-28 00:22:34, 2016-09-27 23:45:00, 201…

## $ in\_reply\_to\_screen\_name <chr> "", "", "", "", "", "", "", "", "", "", "", ""…

## $ in\_reply\_to\_status\_id <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA…

## $ in\_reply\_to\_user\_id <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA…

## $ is\_quote\_status <lgl> FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALS…

## $ lang <chr> "en", "en", "en", "en", "en", "en", "en", "en"…

## $ retweet\_count <int> 218, 2445, 7834, 916, 859, 2181, 1303, 1833, 4…

## $ favorite\_count <int> 651, 5308, 27234, 2542, 2882, 6172, 2849, 4954…

## $ longitude <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA…

## $ latitude <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA…

## $ place\_id <chr> "", "", "", "", "", "", "", "", "", "", "", ""…

## $ place\_full\_name <chr> "", "", "", "", "", "", "", "", "", "", "", ""…

## $ place\_name <chr> "", "", "", "", "", "", "", "", "", "", "", ""…

## $ place\_type <chr> "", "", "", "", "", "", "", "", "", "", "", ""…

## $ place\_country\_code <chr> "", "", "", "", "", "", "", "", "", "", "", ""…

## $ place\_country <chr> "", "", "", "", "", "", "", "", "", "", "", ""…

## $ place\_contained\_within <chr> "", "", "", "", "", "", "", "", "", "", "", ""…

## $ place\_attributes <chr> "", "", "", "", "", "", "", "", "", "", "", ""…

## $ place\_bounding\_box <chr> "", "", "", "", "", "", "", "", "", "", "", ""…

## $ source\_url <chr> "https://studio.twitter.com", "http://twitter.…

## $ truncated <lgl> FALSE, FALSE, FALSE, FALSE, FALSE, TRUE, FALSE…

## $ entities <chr> "{'media': [{'display\_url': 'pic.twitter.com/X…

## $ extended\_entities <chr> "{'media': [{'display\_url': 'pic.twitter.com/X…

There was the need to check whether the involved the use of languages other than English

tweets %>% group\_by(lang) %>% count() %>% rename(Language = lang, 'Number of Tweets' = n)

## # A tibble: 8 × 2

## # Groups: Language [8]

## Language `Number of Tweets`

## <chr> <int>

## 1 da 3

## 2 en 6248

## 3 es 105

## 4 et 1

## 5 fi 1

## 6 fr 2

## 7 tl 2

## 8 und 82

For this project, I will use tweets only in English. From the tweets dataset, the official Twitter name for Donald Trump is reakDonaldTrump, and that of Hillary Clinton is Hillary Clinton. To ensure flexibility and simplicity, the project has shortened these names.

The official Twitter name of Donald Trump is “realDonaldTrump,” and the Twitter name of Hillary Clinton is Hillary Clinton. However, I have shortened their names for this analysis to keep it simple. The table below illustrates the number of non-Spanish tweets for Hillary Clinton and Donald Trump.

tweets <- tweets %>% filter(lang != "es")

tweets$handle <- sub("realDonaldTrump", "Trump", tweets$handle)

tweets$handle <- sub("HillaryClinton", "Clinton", tweets$handle)

tweets$is\_retweet <- as.logical(tweets$is\_retweet)

tweets %>% filter(is\_retweet==FALSE) %>% group\_by(handle) %>% count()

## # A tibble: 2 × 2

## # Groups: handle [2]

## handle n

## <chr> <int>

## 1 Clinton 2557

## 2 Trump 3091

The next thing was to check how many people retweeted tweets by Donald and Hillary. The project illustrates only people that have retweeted at least 5 times. It is clear that there are only 274 people who were retweeted.

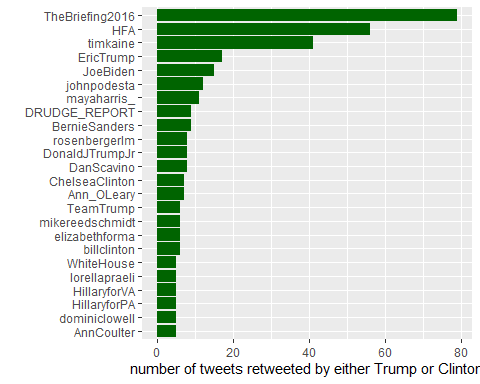
p1 <- tweets %>% filter(original\_author != "") %>% group\_by(original\_author) %>% count() %>% filter(n>=5) %>% arrange(desc(n)) %>% ungroup()

ggplot(p1, aes(x=reorder(original\_author, n), y=n)) +

geom\_bar(stat="identity", fill="darkgreen") + coord\_flip() +

labs(x="", y="number of tweets retweeted by either Trump or Clinton") +

theme(legend.position = "none")



## Methodology

After having identified the tweets from both Hillary and Donald, the next thing was to do the text as shown below.

tweets$author <- ifelse(tweets$original\_author != "", tweets$original\_author, tweets$handle)

head(tweets %>% select(author, handle, text), 20)

## # A tibble: 20 × 3

## author handle text

## <chr> <chr> <chr>

## 1 Clinton Clinton "The question in this election: Who can put the plans into …

## 2 timkaine Clinton "Last night, Donald Trump said not paying taxes was \"\"sma…

## 3 POTUS Clinton "Couldn't be more proud of @HillaryClinton. Her vision and …

## 4 Clinton Clinton "If we stand together, there's nothing we can't do. \n\nMak…

## 5 Clinton Clinton "Both candidates were asked about how they'd confront racia…

## 6 Trump Trump "Join me for a 3pm rally - tomorrow at the Mid-America Cent…

## 7 Clinton Clinton "This election is too important to sit out. Go to https://t…

## 8 Clinton Clinton "When Donald Trump goes low...register to vote: https://t.c…

## 9 Trump Trump "Once again, we will have a government of, by and for the p…

## 10 mcuban Clinton "3) Has Trump offered a single proposal to reduce the frict…

## 11 ddlovato Clinton "The election is just weeks away. Check if you're registere…

## 12 GOP Trump "On National #VoterRegistrationDay, make sure you're regist…

## 13 Trump Trump "Hillary Clinton's Campaign Continues To Make False Claims …

## 14 Trump Trump "'CNBC, Time magazine online polls say Donald Trump won the…

## 15 Clinton Clinton "Donald Trump lied to the American people at least 58 times…

## 16 Trump Trump "Great afternoon in Little Havana with Hispanic community l…

## 17 Trump Trump "In the last 24 hrs. we have raised over $13M from online d…

## 18 Clinton Clinton "“She gained about 55 pounds in...9 months. She was like an…

## 19 Clinton Clinton "It's #NationalVoterRegistrationDay. Celebrate by registeri…

## 20 Clinton Clinton "\"\"I love this country.\nI’m proud of this country.\nI wa…

The text contained some Regex like “”and “"

tweets$text[c(2,4)]

## [1] "Last night, Donald Trump said not paying taxes was \"\"smart.\"\" You know what I call it? Unpatriotic. https://t.co/t0xmBfj7zF"

## [2] "If we stand together, there's nothing we can't do. \n\nMake sure you're ready to vote: https://t.co/tTgeqxNqYm https://t.co/Q3Ymbb7UNy"

In this project, the first thing will involve the removal of as removePunctuation. Also, I will remove the URLs. Next will involve converting the tweet text to ascii to ensure the removal of emojis.

tweets$text <- str\_replace\_all(tweets$text, "[\n]" , "") #remove new lines

tweets$text <- str\_replace\_all(tweets$text, "&amp", "") # rm ampersand

#URLs are always at the end and did not counts towards the 140 characters limit

tweets$text <- str\_replace\_all(tweets$text, "http.\*" , "")

tweets$text <- iconv(tweets$text, "latin1", "ASCII", sub="")

head(tweets)

## # A tibble: 6 × 29

## id handle text is\_retweet original\_author time

## <int64> <chr> <chr> <lgl> <chr> <dttm>

## 1 7e17 Clinton "The question … FALSE "" 2016-09-28 00:22:34

## 2 7e17 Clinton "Last night, D… TRUE "timkaine" 2016-09-27 23:45:00

## 3 7e17 Clinton "Couldn't be m… TRUE "POTUS" 2016-09-27 23:26:40

## 4 7e17 Clinton "If we stand t… FALSE "" 2016-09-27 23:08:41

## 5 7e17 Clinton "Both candidat… FALSE "" 2016-09-27 22:30:27

## 6 7e17 Trump "Join me for a… FALSE "" 2016-09-27 22:13:24

## # … with 23 more variables: in\_reply\_to\_screen\_name <chr>,

## # in\_reply\_to\_status\_id <dbl>, in\_reply\_to\_user\_id <dbl>,

## # is\_quote\_status <lgl>, lang <chr>, retweet\_count <int>,

## # favorite\_count <int>, longitude <dbl>, latitude <dbl>, place\_id <chr>,

## # place\_full\_name <chr>, place\_name <chr>, place\_type <chr>,

## # place\_country\_code <chr>, place\_country <chr>,

## # place\_contained\_within <chr>, place\_attributes <chr>, …

Next will involve the cleaning of the text, where I will remove numbers, strip whitespaces, remove all English stopwords, convert all characters to lower characters, and remove punctuation.

tweets <- tweets %>% rename (doc\_id = id)

ClintonTweets <- tweets %>% filter(is\_retweet=="FALSE" & handle=="Clinton")

TrumpTweets <- tweets %>% filter(is\_retweet=="FALSE" & handle=="Trump")

TrumpCorpus <- DataframeSource(TrumpTweets)

TrumpCorpus <- VCorpus(TrumpCorpus)

ClintonCorpus <- DataframeSource(ClintonTweets)

ClintonCorpus <- VCorpus(ClintonCorpus)

TrumpCorpus

## <<VCorpus>>

## Metadata: corpus specific: 0, document level (indexed): 27

## Content: documents: 3091

CleanCorpus <- function(x){

x <- tm\_map(x, content\_transformer(tolower))

x <- tm\_map(x, removeNumbers) #remove numbers before removing words. Otherwise "trump2016" leaves "trump"

x <- tm\_map(x, removeWords, tidytext::stop\_words$word)

x <- tm\_map(x, removePunctuation)

x <- tm\_map(x, stripWhitespace)

return(x)

}

RemoveNames <- function(x) {

x <- tm\_map(x, removeWords, c("donald", "hillary", "clinton", "trump", "realdonaldtrump", "hillaryclinton"))

return(x)

}

CreateTermsMatrix <- function(x) {

x <- TermDocumentMatrix(x)

x <- as.matrix(x)

y <- rowSums(x)

y <- sort(y, decreasing=TRUE)

return(y)

}

TrumpCorpus <- CleanCorpus(TrumpCorpus)

TermFreqTrump <- CreateTermsMatrix(TrumpCorpus)

content(TrumpCorpus[[1]])

## [1] "join pm rally tomorrow midamerica center council bluffs iowa tickets "

Next, I performed sentiment analysis using the tidytext package that involves several sentiment lexicons

get\_sentiments("bing")

## # A tibble: 6,786 × 2

## word sentiment

## <chr> <chr>

## 1 2-faces negative

## 2 abnormal negative

## 3 abolish negative

## 4 abominable negative

## 5 abominably negative

## 6 abominate negative

## 7 abomination negative

## 8 abort negative

## 9 aborted negative

## 10 aborts negative

## # … with 6,776 more rows

A wordcloud has been created to help identify the most used words by Trump, the Republican candidate.

set.seed(2018)

TrumpCorpus1 <- RemoveNames(TrumpCorpus)

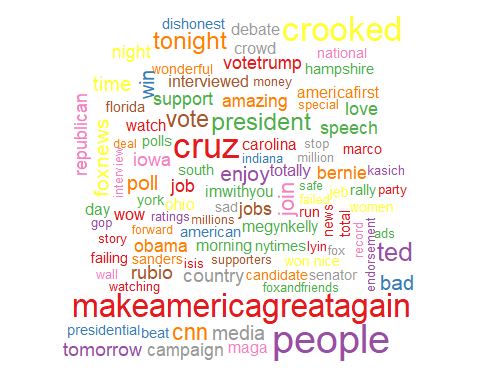
TermFreqTrump <- CreateTermsMatrix(TrumpCorpus1)

TrumpDF <- data.frame(word=names(TermFreqTrump), count=TermFreqTrump)

wordcloud(TrumpDF$word, TrumpDF$count, max.words = 100, scale=c(2.5,.5), random.color = TRUE, colors=brewer.pal(9,"Set1"))

## Warning in wordcloud(TrumpDF$word, TrumpDF$count, max.words = 100, scale =

## c(2.5, : america could not be fit on page. It will not be plotted.



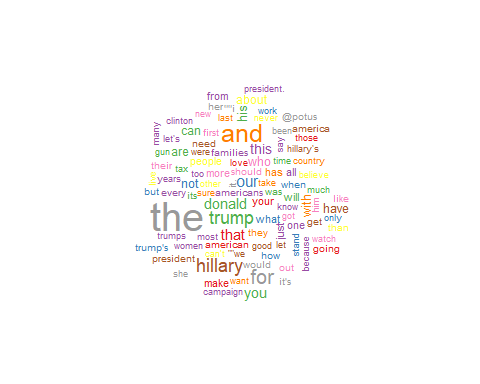
A wordcloud has been created to help identify the most used words by Clinton, the Democrat presidential candidate.

ClintonCorpus1 <- RemoveNames(ClintonCorpus)

TermFreqClinton <- CreateTermsMatrix(ClintonCorpus1)

ClintonDF <- data.frame(word=names(TermFreqClinton), count=TermFreqClinton)

wordcloud(ClintonDF$word, ClintonDF$count, max.words = 100, scale=c(2.5,.5), random.color = TRUE, colors=brewer.pal(9,"Set1"))



## Results

From the analysis performed in this project, it is illustrated that the trump was the most used term and enjoyed was the least used term.

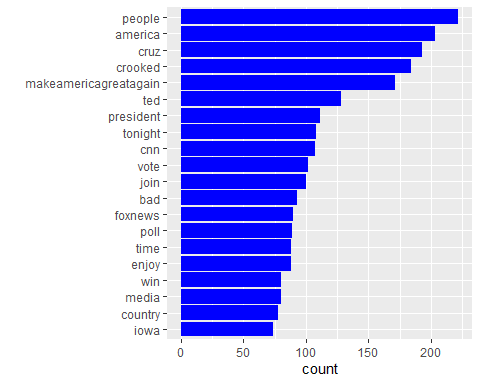
TrumpDF <- data.frame(word=names(TermFreqTrump), count=TermFreqTrump)

TrumpDF[1:20,] %>%

ggplot(aes(x=(reorder(word, count)), y=count)) +

geom\_bar(stat='identity', fill="blue") + coord\_flip() + theme(legend.position = "none") +

labs(x="")



To understand the difference of words used by the two presidential candidates in their tweets, I created the comparison cloud using the function comparison.cloud. Hillary clinton as identified used the word president more often unlike Trump.

allClinton <- paste(ClintonTweets$text, collapse = " ")

allTrump <- paste(TrumpTweets$text, collapse = " ")

allClTr <- c(allClinton, allTrump)

allClTr <- VectorSource(allClTr)

allCorpus <- VCorpus(allClTr)

allCorpus <- CleanCorpus(allCorpus)

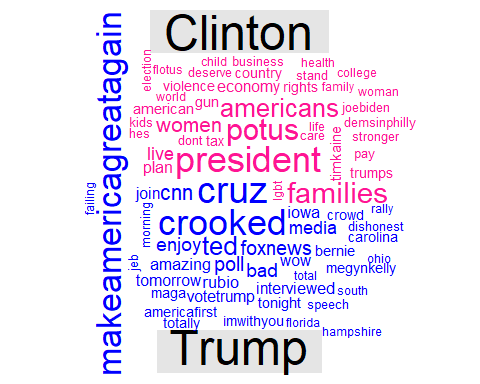
allCorpus <- RemoveNames(allCorpus)

TermsAll <- TermDocumentMatrix(allCorpus)

colnames(TermsAll) <- c("Clinton", "Trump")

MatrixAll <- as.matrix(TermsAll)

comparison.cloud(MatrixAll, colors = c("#FF1493", "blue"), scale=c(2.3,.3), max.words = 75)



TrumpTidy <- tidy(TrumpCorpus)

ClintonTidy <- tidy(ClintonCorpus)

TrumpTidy1 <- tidy(TrumpCorpus1) #without names

ClintonTidy1 <- tidy(ClintonCorpus1) #without names

##adding the date of the Tweets from the document level meta data

DocMetaTrump1 <- meta(TrumpCorpus1)

DocMetaTrump1$date <- date(DocMetaTrump1$time)

TrumpTidy1$date <- DocMetaTrump1$date

DocMetaClinton1 <- meta(ClintonCorpus1)

DocMetaClinton1$date <- date(DocMetaClinton1$time)

ClintonTidy1$date <- DocMetaClinton1$date

NoNamesTidy <- bind\_rows(trump=TrumpTidy1, clinton=ClintonTidy1, .id="candidate")

Words <- NoNamesTidy %>% unnest\_tokens(word, text)

## Warning: Outer names are only allowed for unnamed scalar atomic inputs

Bing <- Words %>% inner\_join(get\_sentiments("bing"), by="word")

b1 <- Bing %>% filter(candidate=="trump") %>% count(word, sentiment, sort=TRUE) %>%

group\_by(sentiment) %>% arrange(desc(n)) %>% slice(1:20) %>%

ggplot(aes(x=reorder(word, n), y=n)) +

geom\_col(aes(fill=sentiment), show.legend=FALSE) +

coord\_flip() +

facet\_wrap(~sentiment, scales="free\_y") +

labs(x="", y="number of times used", title="Donald Trump's most used words") +

scale\_fill\_manual(values = c("positive"="green", "negative"="red"))

b2 <- Bing %>% filter(candidate=="clinton") %>% count(word, sentiment, sort=TRUE) %>%

group\_by(sentiment) %>% arrange(desc(n)) %>% slice(1:20) %>%

ggplot(aes(x=reorder(word, n), y=n)) +

geom\_col(aes(fill=sentiment), show.legend=FALSE) +

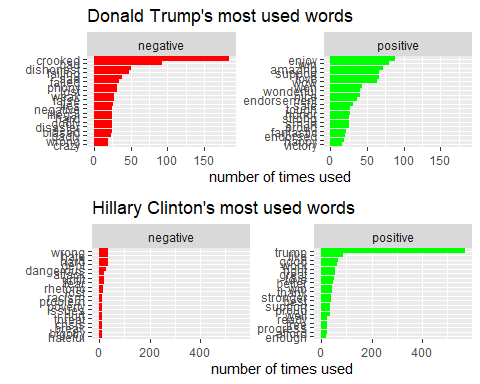
coord\_flip() +

facet\_wrap(~sentiment, scales="free\_y") +

labs(x="", y="number of times used", title="Hillary Clinton's most used words") +

scale\_fill\_manual(values = c("positive"="green", "negative"="red"))

grid.arrange(b1, b2)



Through the use of the nrc lexicon. the words were categorized into positive or negative sentiment as well as into 8 basic emotions, that is, trust, anticipation, joy, anger, surprise, fear, sadness, and disgust.

get\_sentiments("nrc")

## # A tibble: 13,872 × 2

## word sentiment

## <chr> <chr>

## 1 abacus trust

## 2 abandon fear

## 3 abandon negative

## 4 abandon sadness

## 5 abandoned anger

## 6 abandoned fear

## 7 abandoned negative

## 8 abandoned sadness

## 9 abandonment anger

## 10 abandonment fear

## # … with 13,862 more rows

Through the use of the nrc lexicon, it was possible to identify the difference between the sentiments in Trump's and Hillary's tweets. It was identified that Trump used more negativity as well as a bit of disgust, anger, and sadness.

Nrc <- Words %>% inner\_join(get\_sentiments("nrc"), by="word")

n1 <- Nrc %>% filter(candidate=="trump") %>% count(sentiment) %>%

ggplot(aes(x=sentiment, y=n, fill=sentiment)) +

geom\_bar(stat="identity") + coord\_polar() +

theme(legend.position = "none", axis.text.x = element\_blank()) +

geom\_text(aes(label=sentiment, y=2500)) +

labs(x="", y="", title="Trump")

n2 <- Nrc %>% filter(candidate=="clinton") %>% count(sentiment) %>%

ggplot(aes(x=sentiment, y=n, fill=sentiment)) +

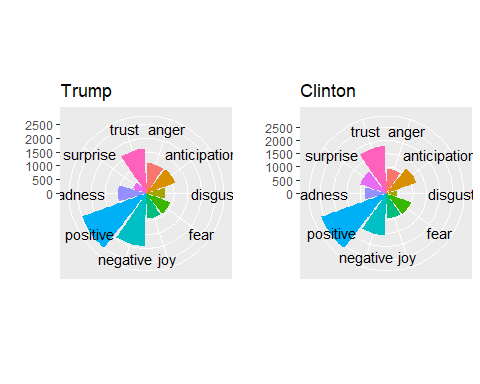
geom\_bar(stat="identity") + coord\_polar() +

theme(legend.position = "none", axis.text.x = element\_blank()) +

geom\_text(aes(label=sentiment, y=2500)) +

labs(x="", y="", title="Clinton")

grid.arrange(n1, n2, nrow=1)



## Conclusion

The presidential candidates, in their tweets, expressed different sentiments and emotions. The Democrat candidate Hillary Clinton’s tweets involve positive sentiments and emotions. However, the tweets from the Republican candidate, Donald Trump, involve more negative sentiments and emotions and use more negation words is true. This is because, from the analysis performed in this project, it is clear that Trump had more negativity than Clinton in their tweets.

# References

Calvin, A. J., Bellmore, A., Xu, J. M., & Zhu, X. (2015). # bully: Uses of hashtags in posts about bullying on Twitter. Journal of School Violence, 14(1), 133-153.

Jahanbakhsh, K., & Moon, Y. (2014). The predictive power of social media: On the predictability of us presidential elections using twitter. arXiv preprint arXiv:1407.0622.

Joyce, B., & Deng, J. (2017, November). Sentiment analysis of tweets for the 2016 US presidential election. In 2017 ieee mit undergraduate research technology conference (urtc) (pp. 1-4). IEEE.

Knight, C. G., & Kaye, L. K. (2016). ‘To tweet or not to tweet?’A comparison of academics’ and students’ usage of Twitter in academic contexts. Innovations in education and teaching international, 53(2), 145-155.

Wang, W., Chen, L., Thirunarayan, K., & Sheth, A. P. (2012, September). Harnessing twitter” big data” for automatic emotion identification. In 2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing (pp. 587-592). IEEE.