TWITTER TEXT MINING

January 31st, 2017

INTRO

Twitter mining is usually known for its ability to analyze social campaigns or track public sentiment for a specific company or organization.

But what about movements?

In response to one Donald Trump's first major executive order in office, relating to the ban of refugees, immigration and green card holders/dual citizenship owners from 'radical Islamic' countries, specifically Syria, Somalia, Sudan, Iran, Iraq, Libya and Yemen, two major twitter movements erupted. These movements represent the disapproval and insurgence of two disparate political groups and are namely described by the twitter hashtags #BoycottUber and #BoycottStarbucks.

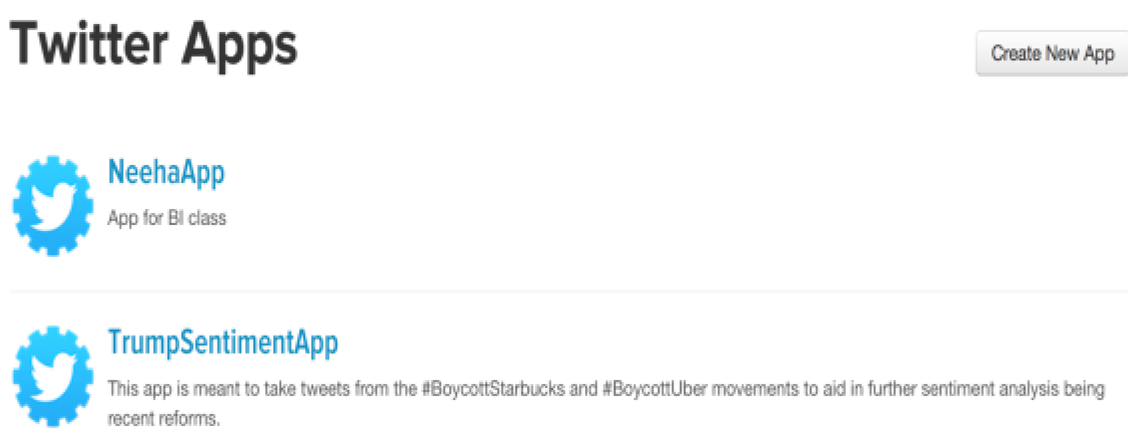
Though both movements were sparked by the abrupt and controversial orders of president Donald Trump, they represent sparsely different political esteems.

My goal in performing this analysis is to analyze major motives, popular phrases, and sentiment using Wordcloud analysis and text-mining in a quantitative and unbiased manner.

TWITTER AUTHORIZATION

Twitter Authorization consisted of

a) Creating a twitter account and
b) Creating an app on twitter



c) Configuring twitter app account with

Application Settings

Keep the "Consumer Secret" a secret. This key should never be human-readable in your application.

Consumer Key (API Key)	
Consumer Secret (API Secret)	
Access Level	Read and write (modify app permissions)
Owner	NeehaKaja
Owner ID	

Application Actions

Regenerate Consumer Key and Secret | Change App Permissions

Your Access Token

This access token can be used to make API requests on your own account's behalf. Do not share your access token secret with anyone.

Access Token	
Access Token Secret	
Access Level	Read and write
Owner	NeehaKaja
Owner ID	

d) Running setup_twitter_oath method.

```
consumer_key <- "twitter consumer key here"
consumer_secret <- "twitter consumer secret here"
access_token <- "twitter access token here"
access_secret <- "twitter access secret here"
setup_twitter_oauth(consumer_key, consumer_secret, access_token, access_secret)
```

UBERSCRIPT.R
neeharikakaja

Mon Jan 30 22:35:51 2017

```
## [1] "Using direct authentication"

boycott_uber <- searchTwitter("#BoycottUber", n=1500)
boycott_uber_df <- do.call("rbind", lapply(boycott_uber, as.data.frame))
boycott_uber_tweets <- boycott_uber_df$text
#convert tweets to utf to get rid of unknown error with TDM/gets rid of bad ch
aracters
boycott_uber_tweets <- iconv(boycott_uber_tweets,to="utf-8-mac")

boycott_uber_source <- VectorSource(boycott_uber_tweets)
boycott_uber_corpus <- VCorpus(boycott_uber_source)
boycott_uber_corpus

## <<VCorpus>>
## Metadata:  corpus specific: 0, document level (indexed): 0
## Content:  documents: 1500

#extent of cleaning - do better job/especially takeout common words
boycott_uber_corpus <- tm_map(boycott_uber_corpus,removePunctuation)
boycott_uber_corpus <- tm_map(boycott_uber_corpus,stripWhitespace)
boycott_uber_corpus <- tm_map(boycott_uber_corpus,removePunctuation)
boycott_uber_corpus <- tm_map(boycott_uber_corpus, removeWords, c(stopwords("e
n"), "the"))


boycott_uber_tdm <- TermDocumentMatrix(boycott_uber_corpus)
boycott_uber_m <- as.matrix(boycott_uber_tdm)


dim(boycott_uber_m)

## [1] 2628 1500

term_frequency <- rowSums(boycott_uber_m)
term_frequency <- sort(term_frequency, decreasing = TRUE)
#testing purp
#term_frequency[1:10]
#barplot(term_frequency[1:10], col="tan", las=2)


word_freqs <- data.frame(term = names(term_frequency), num=term_frequency)

# Create a wordcloud for the values in word_freqs
wordcloud(word_freqs$term, word_freqs$num, max.words=50, colors = "blue", min.
freq = 1, scale=c(5,.3))
```

Process:

In this Wordcloud I use the twitter package in R and the method searchTwitter
to pull 1500 #BoycottUber tweets via the public Twitter API for further analys
is.

I then go through a few steps of cleaning the data by converting it to a data
frame, Vector Source, Volatile Corpus, clean text, Term Document Matrix (this
matrix places each term in a row and each tweet in a column, 2628 terms and 15
00 columns from a list.

I then compute the row sums of the Term Document Matrix to understand the freq
uency that each term appears in my matrix (frequency will be the determining f
actor for plotting the Wordcloud) and store this information in a data frame t
hat can be used by the Wordcloud function.

The Wordcloud function plots terms by frequency in a blue color the represent
the more ‘liberal’ views of the “Boycott Uber” movement and scales it.

Analysis:

In analyzing my graph I see that the following words stand out: uber, deleteub
er, trump, London, lyft, boycottstarbucks, taxi, trade, support, savetaxi, thi
ef, and whateverceo.

```
The rhetoric seems to suggest that those who support “Boycott Uber” seem to fe
el strongly about Donald Trump, deleting uber, perhaps adopting lyft or taxi s
ervices, reference the boycott starbucks movements, and view Uber’s ceo as a
‘thief’.

These outcomes were expected.
```

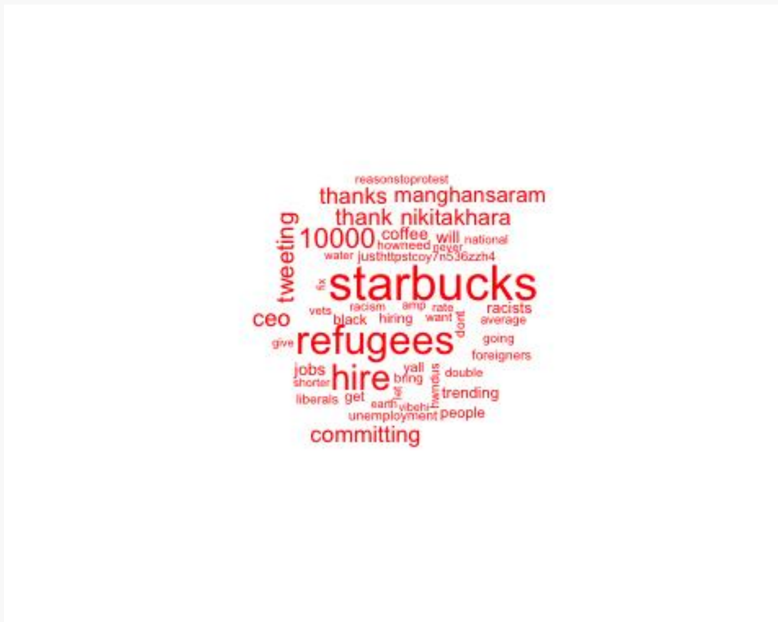
STARBUCKSSCRIPT.R

neeharikakaja

```
## [1] "Using direct authentication"

boycott_starbucks <- searchTwitter("#boycottstarbucks", n=1500)
boycott_starbucks_df <- do.call("rbind", lapply(boycott_starbucks, as.data.fra
me))
boycott_starbucks_tweets <- boycott_starbucks_df$text
#convert tweets to utf to get rid of unknown error with TDM/gets rid of bad ch
aracters
boycott_starbucks_tweets <- iconv(boycott_starbucks_tweets,to="utf-8-mac")
boycott_starbucks_tweets <- iconv(boycott_starbucks_tweets, "latin1", "ASCII",
sub="")


boycott_starbucks_source <- VectorSource(boycott_starbucks_tweets)
boycott_starbucks_corpus <- VCorpus(boycott_starbucks_source)
boycott_starbucks_corpus

## <<VCorpus>>
## Metadata:  corpus specific: 0, document level (indexed): 0
## Content:  documents: 1500

#extent of cleaning - do better job/especially takeout common words
boycott_starbucks_corpus <- tm_map(boycott_starbucks_corpus,removePunctuation)
boycott_starbucks_corpus <- tm_map(boycott_starbucks_corpus,stripWhitespace)
boycott_starbucks_corpus <- tm_map(boycott_starbucks_corpus,removePunctuation)
boycott_starbucks_corpus <- tm_map(boycott_starbucks_corpus, removeWords, c(st
opwords("en"), "the"))


boycott_starbucks_tdm <- TermDocumentMatrix(boycott_starbucks_corpus)
boycott_starbucks_m <- as.matrix(boycott_starbucks_tdm)


term_frequency <- rowSums(boycott_starbucks_m)
term_frequency <- sort(term_frequency, decreasing = TRUE)


#testing purp
#term_frequency[1:10]
#barplot(term_frequency[1:10], col="tan", las=2)


word_freqs <- data.frame(term = names(term_frequency), num=term_frequency)
```

```r
# Create a wordcloud for the values in word_freqs
wordcloud(word_freqs$term, word_freqs$num, max.words=50, colors = "red", min.freq = 1, scale=c(5,.3))
```

```
## Warning in wordcloud(word_freqs$term, word_freqs$num, max.words = 50,
## colors = "red", : boycottstarbucks could not be fit on page. It will not be
## plotted.
```



Process:

In this Wordcloud I use the twitter package in R and the method searchTwitter to pull 1500 #BoycottUber tweets via the public Twitter API for further analysis.

I then go through a few steps of cleaning the data by converting it to a data frame, Vector Source, Volatile Corpus, clean text, Term Document Matrix (this matrix places each term in a row and each tweet in a column, 2628 terms and 1500 columns from a list.

I then compute the row ums of the Term Document Matrix to understand the frequency that each term appears in my matrix (frequency will be the determining factor for plotting the Wordcloud) and store this information in a data frame that can be used by the Wordcloud function.

The Wordcloud function plots terms by frequency in a blue color to represent the more ‘liberal’ views of the “Boycott Uber” movement and scales it.

Analysis:

```
In this graph I see that the following words stand out: starbucks, refugees, h
ire, committing, ceo, racists, foreigners, black, liberals, national, vets, jo
bs, and reasonstoprotest.



This rhetoric suggests that the Boycott Starbucks movement is much more politi
cally (referencing liberals), racially (referencing blacks foreigners), econom
ically (jobs, hire), and regionally (foreigners, nationals) than the Boycott U
ber movement. The Boycott Uber movement focusses more on alternatives to Uber
(lyft, taxis) and the disapproval of Uber's profiting during the ban (dollars,
whateverceo etc).

These outcomes were expected.
```

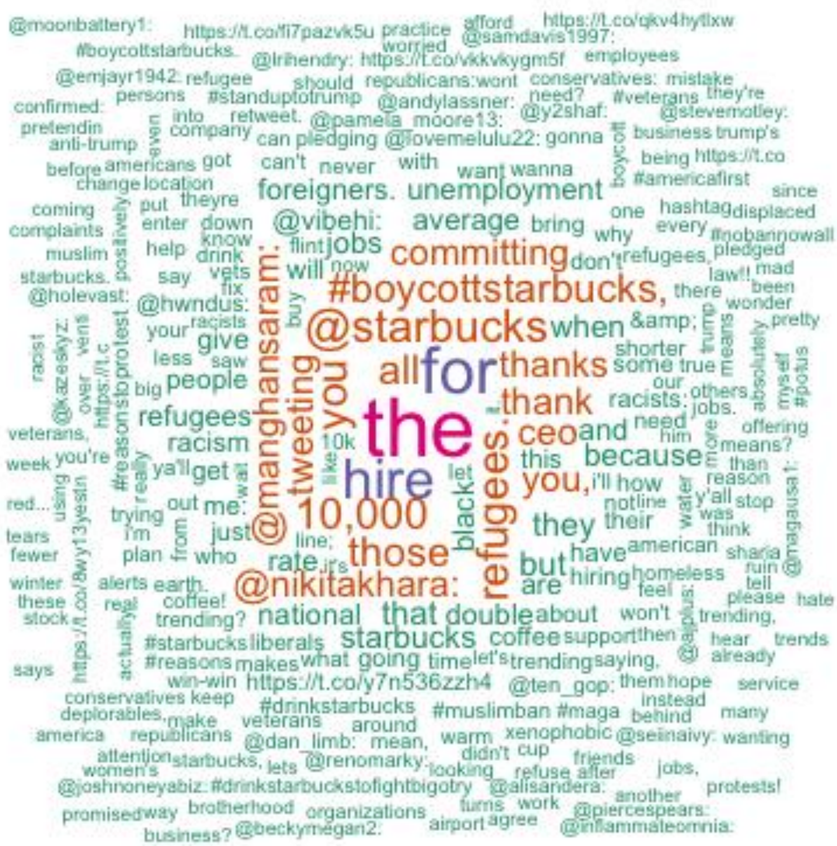
COMMONSCRIPT.R

neeharikakaja

Tue Jan 31 00:20:33 2017

```
## [1] "Using direct authentication"

#get starbucks tweets
boycott_starbucks <- searchTwitter("#boycottstarbucks", n=1500)
boycott_starbucks_df <- do.call("rbind", lapply(boycott_starbucks, as.data.fra
me))
boycott_starbucks_tweets <- boycott_starbucks_df$text
#convert tweets to utf to get rid of unknown error with TDM/gets rid of bad ch
aracters
boycott_starbucks_tweets <- iconv(boycott_starbucks_tweets,to="utf-8-mac")
boycott_starbucks_tweets <- iconv(boycott_starbucks_tweets, "latin1", "ASCII",
sub="")

#get starbucks tweets
boycott_uber <- searchTwitter("#boycottstarbucks", n=1500)
boycott_uber_df <- do.call("rbind", lapply(boycott_uber, as.data.frame))
boycott_uber_tweets <- boycott_uber_df$text
#convert tweets to utf to get rid of unknown error with TDM/gets rid of bad ch
aracters
boycott_uber_tweets <- iconv(boycott_uber_tweets,to="utf-8-mac")
boycott_uber_tweets <- iconv(boycott_uber_tweets, "latin1", "ASCII", sub="")



all_uber <- paste(boycott_uber_tweets, collapse = " ")
all_starbucks <- paste(boycott_starbucks_tweets, collapse = " ")

all_tweets <- c(all_uber, all_starbucks)
all_tweets <- VectorSource(all_tweets)
all_corpus <- VCorpus(all_tweets)
```

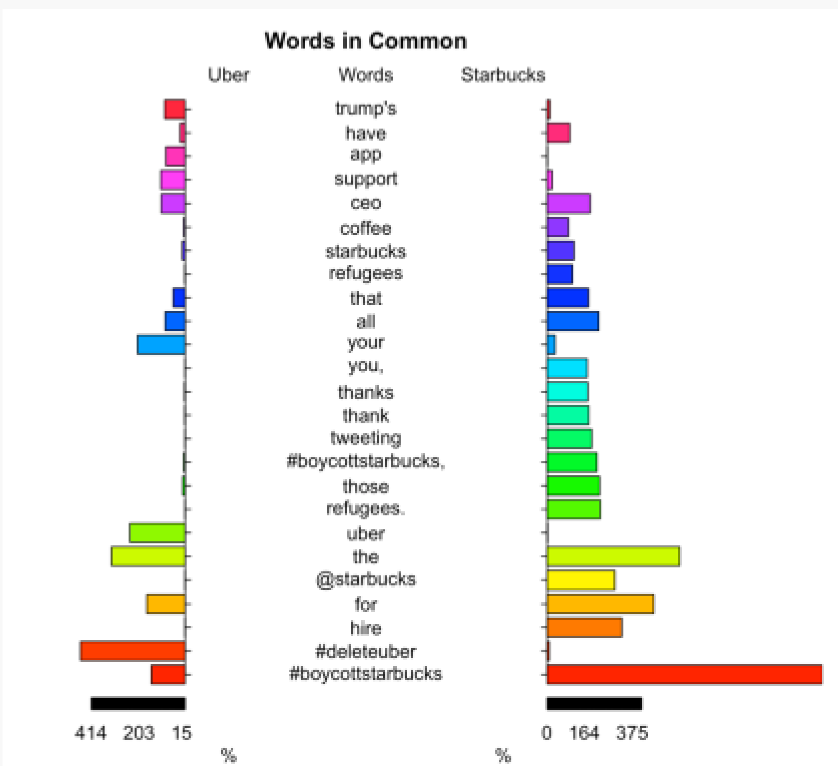
```
all_clean <- all_corpus
all_tdm <- TermDocumentMatrix(all_clean)
colnames(all_tdm) <- c("Uber", "Starbucks")
all_m <- as.matrix(all_tdm)

#common cloud
commonality.cloud(all_m, random.order=FALSE,
                  colors = brewer.pal(8, "Dark2"),
                  title.size=1.5)
```




```
Process:

In this Wordcloud I collapsed both the Boycott Uber tweets and Boycott Starbuc
ks tweets into one common matrix, converted them into a Vector Source, Volatil
e Corpus, Term Document Matrix and Matrix for processing and then plotted them
with the commonality.cloud function available in R for visualization.



The purpose of collapsing the terms was to see what terms occur most frequentl
y between both movements and identify points of commonality.
```

```
Analysis:

In this graph I see that the following words stand out: boycottstarbucks, than
k you, refugees, unemployment, foreigners, racism, American, black, homeless,
help, and pledging, suggesting that both movements are likely addressing aspec
ts of racial and economic issues (foreigners, racism, black, American, refugee
s, unemployment, homeless) in a passionate passive manner (pledging, thank yo
u).
```

```r
#polarized tag cloud
common_words <- subset(all_m, all_m[, 1] > 0 & all_m[, 2] > 0)
difference <- abs(common_words[, 1] - common_words[, 2])
common_words <- cbind(common_words, difference)
common_words <- common_words[order(common_words[, 3], decreasing = TRUE), ]

# Create top25_df
top25_df <- data.frame(x = common_words[1:25, 1],
                       y = common_words[1:25, 2],
                       labels = rownames(common_words[1:25, ]))


library(plotrix)
#Create the pyramid plot
pyramid.plot(top25_df$x, top25_df$y, labels = top25_df$labels,
             gap = 800, top.labels = c("Uber", "Words", "Starbucks"),
             main = "Words in Common")
```



Process:

According to Data Camp, the website I used to guide my analysis process, the commonality.cloud() may be misleading because words can be represented disproportionately in one corpus or the other, even if they are shared. To solve this problem we can create a pyramid.plot() from the plotrix package in R.

I subsetted the common words using R's subset function, found the absolute difference between common words, ordered them from greatest to least (this would allow us to pinpoint which common words occur in the most polarizing quantities in either movement), and plotted them via the pyramid.plot() function in R.

Analysis:

In the graph we can see that the most polarizing terms occur in red near the bottom of the graph, "#boycottstarbucks" occurring more frequency in the Boycott Starbucks movement and "#deleteuber occurring more frequently in the Boycott Uber movement.

This was expected and serves as a qualifying feature, suggesting that the data and plot is working as expected and is reliable.

Next, we see that terms like refugees, thank you, and ceo seem to appear more proportionately in the Boycott Starbucks movement. This suggests that the Boycott Starbucks movement focusses more on their disapproval of refugees than other topics such as Donald Trump, or Ceo's. It also suggests that perhaps the Boycott Starbucks movement is more celebratory in nature, using terms like 'thank you'.

Contrastingly, the Boycott Uber movement references trump and "app"[s] in greater proportion than the Boycott Starbucks movement, suggesting that the Boycott Uber movement focusses more its disapproval of Trump and deleting the Uber app than other topics such as refugees or economic factors.

These results were by far the most surprising of my analysis and provides the greatest incite about both movements.

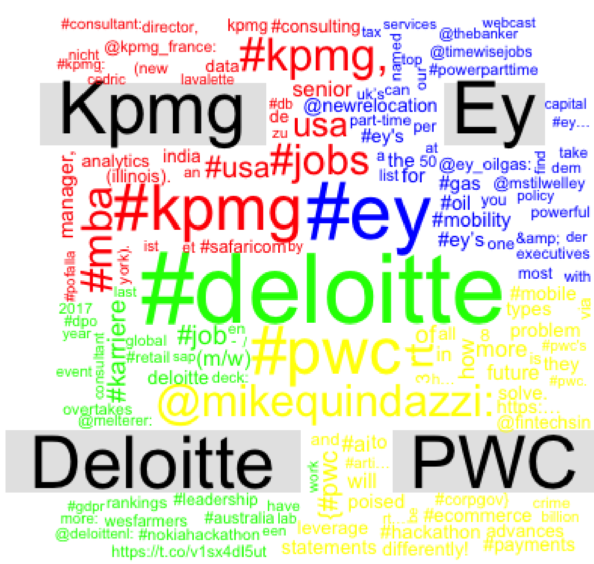
BIG FOUR ANALYSIS

After seeing the success of my Boycott Uber and Boycott Starbucks analysis, I was curious to see the effect of R text mining on one more subject relevant to college students in the MIS major: Reputation of the Big 4.

I mined 500 texts from each of the big four twitter hashtags (#ey, #kpmg, #deloitte, and #pwc) to see whether the general reputations about each company are true.

Below are my results

```
## [1] "Using direct authentication"

#ey
ey <- searchTwitter("#EY", n=500)
ey_df <- do.call("rbind", lapply(ey, as.data.frame))
ey_tweets <- ey_df$text
ey_tweets <- iconv(ey_tweets,to="utf-8-mac")
all_ey <- paste(ey_tweets, collapse = " ")
#kpmg
kpmg <- searchTwitter("#kpmg", n=500)
kpmg_df <- do.call("rbind", lapply(kpmg, as.data.frame))
kpmg_tweets <- kpmg_df$text
kpmg_tweets <- iconv(kpmg_tweets,to="utf-8-mac")
all_kpmg <- paste(kpmg_tweets, collapse = " ")
#deloitte
deloitte <- searchTwitter("#deloitte", n=500)
deloitte_df <- do.call("rbind", lapply(deloitte, as.data.frame))
deloitte_tweets <- deloitte_df$text
deloitte_tweets <- iconv(deloitte_tweets,to="utf-8-mac")
all_deloitte <- paste(deloitte_tweets, collapse = " ")
#pwc
pwc <- searchTwitter("#pwc", n=500)
pwc_df <- do.call("rbind", lapply(pwc, as.data.frame))
pwc_tweets <- pwc_df$text
pwc_tweets <- iconv(pwc_tweets,to="utf-8-mac")
```




```
Analysis:

KPMG comes across as very global and position-growth driven as evidenced by te
rms such as “usa, india, kpmg_france, Illinois, consulting, director, manager,
mba, and jobs”
```

```
EY positions itself as an industrial and media leader with terms like “oil, mo
bility, gas, powerparttime, policy, the 50 list, thebanker, top, and most”.


Deloitte creates a vision for success, leadership and work using terms such as
“rankings, leadership, overtake, event, global, job and work”.


Finally, PWC creates a balanced and cordial agenda with terms like “poised, di
fferently, solve, problem, future, leverage, and mobile.”


I believe that these representations fit the stereotypes of each company and w
ere expected.
```

REFLECTION

```
I enjoyed completing this project and was excited to put the data/text mining
skills I learned from my semester in Business Intelligence at UGA to use.

I was also excited to employ some of the new graphs and packages in R that I h
ad not yet played with (such as plotrix) in analysis of current events (Donald
Trump’s immigrant ban) and feel positively about using these methods of analys
is on other data sets.

A major impediment of my project was the retrieval and display of common stopw
ords/transitory words like ‘is, the, for, your, have,’ in my graphs, suggestin
g that I wasn’t able to clean the data effectively.

The inclusion of common English words in my analysis also may have prevented m
e from gaining deeper insight from my data and it is something I would pay att
ention to more closely in the future (text cleaning and removing stopwords).

Despite this being the case, I was very satisfied with my results because in t
he more advanced graphs, specifically the pyramid plot where was able to under
stand the main difference between the Boycott Uber and Boycott Starbucks movem
ents. Uber’s movement is more greatly focused on criticizing Trump while Starb
ucks’s is focused on other political factors.


Credit: Credit for method of analysis and guidance in learning R attributed to
www.datacamp.com

Data Camp is a great source for students to learn BI and data mining methodolo
gies in R and Python and I would certainly suggest it to others.
```