# Tweet Analysis

*Ashlyn Jew*

*5/14/2020*

Text and sentiment analysis of tweets including keywords "anti-asian" and "virus" (scraped from Jan 24th to May 10th).

## Tokenize text, remove irrelevant words, get count of each word

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(tidytext)
```

```
## Warning: package 'tidytext' was built under R version 3.6.3
```

```
antiasian <- read_csv("anti-asian-3.csv")
```

```
## Warning: Missing column names filled in: 'X1' [1]

## Parsed with column specification:
## cols(
##   .default = col_character(),
##   X1 = col_double(),
##   X = col_double(),
##   user_id = col_double(),
##   tweet_id = col_double(),
##   timestamp = col_datetime(format = ""),
##   timestamp_epochs = col_double(),
##   has_media = col_logical(),
##   video_url = col_logical(),
##   likes = col_double(),
##   retweets = col_double(),
##   replies = col_double(),
##   is_replied = col_logical(),
##   is_reply_to = col_logical(),
##   parent_tweet_id = col_logical(),
##   `data$tweet_id` = col_double()
## )
## See spec(...) for full column specifications.
```

```
#without stop words
my_stop_words <- stop_words %>% select(-lexicon) %>%
  bind_rows(data.frame(word = c("asian", "anti", "virus", "https", "twitter.com", "status", "19", "pic.
```

```
## Warning in bind_rows_(x, .id): binding character and factor vector,
## coercing into character vector
```

```
ordered_text_count <- antiasian %>%
  select(text) %>%
  unnest_tokens(word,text) %>%
  add_count(word) %>%
  distinct() %>%
  anti_join(my_stop_words) %>%
  arrange(desc(n))
```

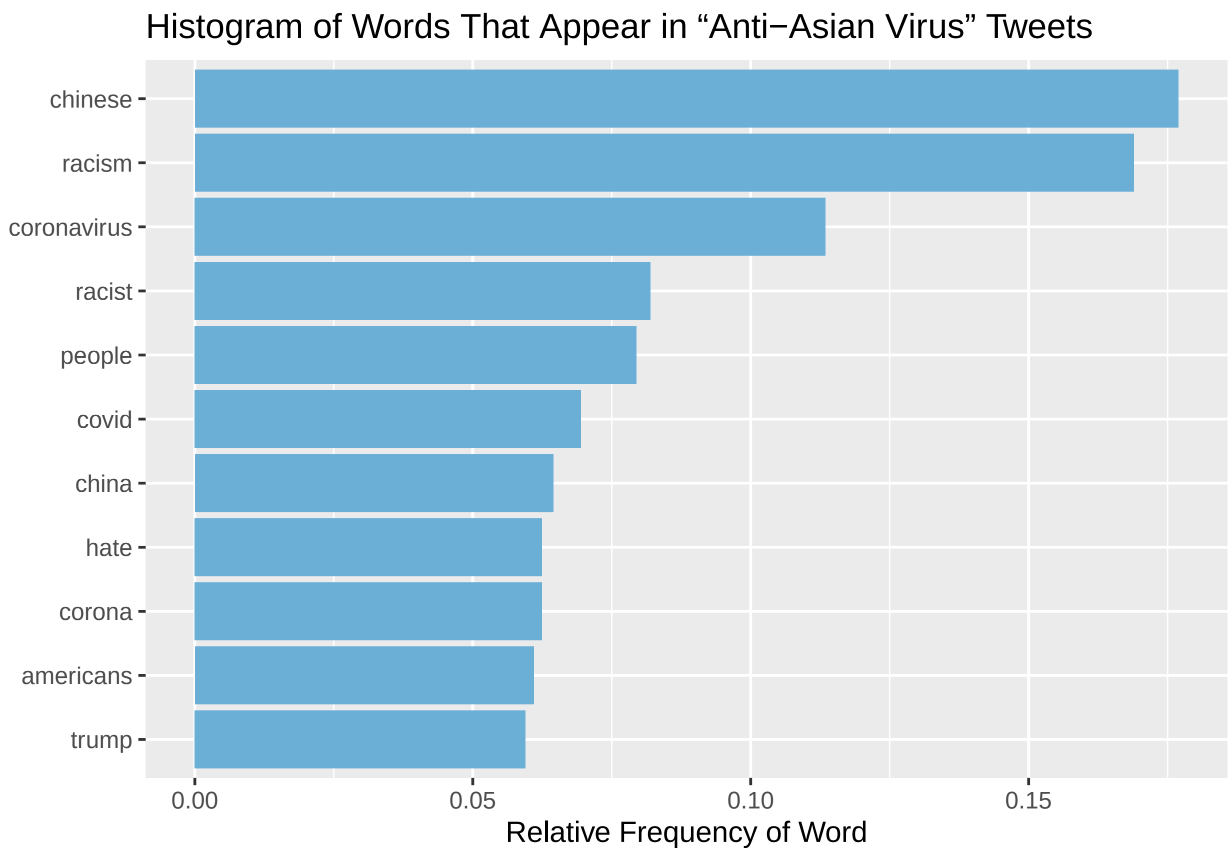
```
## Joining, by = "word"
```

```
ordered_text_count
```

```
## # A tibble: 4,848 x 2
##    word            n
##    <chr>       <int>
##  1 chinese       354
##  2 racism        338
##  3 coronavirus   227
##  4 racist        164
##  5 people        159
##  6 covid         139
##  7 china         129
##  8 corona        125
##  9 hate          125
## 10 americans     122
## # ... with 4,838 more rows
```

## Histogram of Words That Appear in “Anti-Asian Virus” Tweets

```
ordered_text_count %>%
  filter(n > 110) %>%
  mutate(rel_freq = n/sum(n)) %>%
  mutate(word = reorder(word, rel_freq)) %>%
  ggplot(aes(word, rel_freq)) +
  geom_col(fill = "#6baed6") +
  xlab(NULL) +
  ylab("Relative Frequency of Word") +
  ggtitle("Histogram of Words That Appear in "Anti-Asian Virus" Tweets")+
  coord_flip()
```



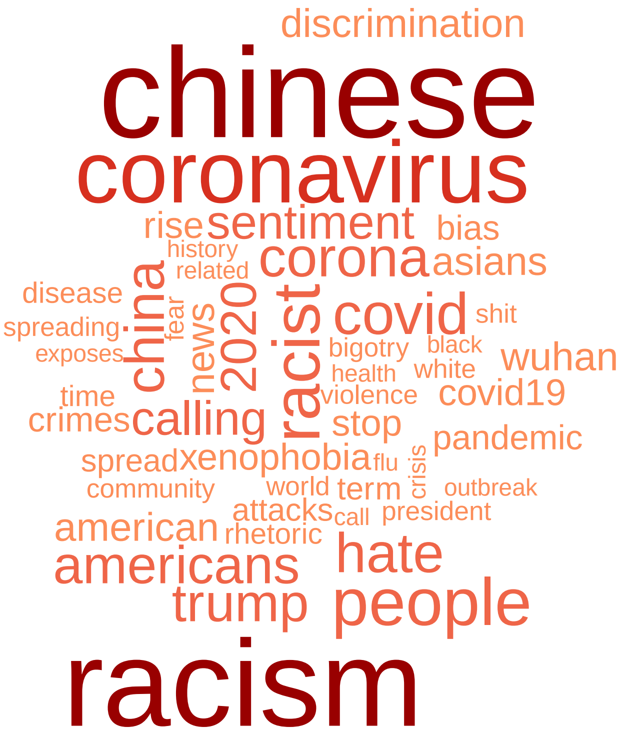
## Wordcloud

```
library(wordcloud)
```

```
## Warning: package 'wordcloud' was built under R version 3.6.3
```

```
## Loading required package: RColorBrewer
```

```
wordcloud(ordered_text_count$word, freq = ordered_text_count$n, max.words = 50, colors = c("#fc8d59", "#
```



## Assigning different sentiment values

```r
library(textdata)
```

```
## Warning: package 'textdata' was built under R version 3.6.3
```

```r
#assigns numeric values
afinn_sent <- get_sentiments("afinn")

#negative or positive
bing_sent <- get_sentiments("bing")

#emotion
nrc_sent <- get_sentiments("nrc")

#emotion words
ordered_text_count %>%
  left_join(nrc_sent) %>%
  filter(!is.na(sentiment), word != "trump")
```

```
## Joining, by = "word"
## # A tibble: 1,822 x 3
##    word                n sentiment
##    <chr>           <int> <chr>
##  1 hate              125 anger
##  2 hate              125 disgust
##  3 hate              125 fear
##  4 hate              125 negative
##  5 hate              125 sadness
##  6 discrimination     79 anger
##  7 discrimination     79 disgust
##  8 discrimination     79 fear
##  9 discrimination     79 negative
## 10 discrimination     79 sadness
## # ... with 1,812 more rows
```

```
#summary of emotion words
 emotion_summary <- ordered_text_count %>%
  left_join(nrc_sent) %>%
  filter(!is.na(sentiment), word != "trump") %>%
  group_by(sentiment) %>%
  summarise(n = n())
```

```
## Joining, by = "word"
```

```
 emotion_summary
```

```
## # A tibble: 10 x 2
##    sentiment       n
##    <chr>       <int>
##  1 anger         196
##  2 anticipation  114
##  3 disgust       143
##  4 fear          217
##  5 joy            80
##  6 negative      385
##  7 positive      300
##  8 sadness       151
##  9 surprise       70
## 10 trust         166
```

```
 #affin
 ordered_text_count %>%
   left_join(afinn_sent) %>%
   filter(!is.na(value)) %>%
   summarise(mean_value = mean(value))
```

```
## Joining, by = "word"
```

```
## # A tibble: 1 x 1
##   mean_value
##        <dbl>
## 1     -0.951
```

```
 #bing
  bing_summary <- ordered_text_count %>%
   left_join(bing_sent) %>%
   filter(!is.na(sentiment), word != "trump") %>%
  group_by(sentiment)
```

```
## Joining, by = "word"
```

```
  bing_summary
```

```
## # A tibble: 586 x 3
## # Groups:   sentiment [2]
##    word               n sentiment
##    <chr>          <int> <chr>
##  1 racism           338 negative
##  2 racist           164 negative
##  3 hate             125 negative
##  4 discrimination    79 negative
##  5 bias              63 negative
##  6 attacks           48 negative
##  7 rhetoric          45 negative
##  8 bigotry           33 negative
##  9 fear              32 negative
## 10 shit              30 negative
## # ... with 576 more rows
```

## Histogram of Tweet Sentiments

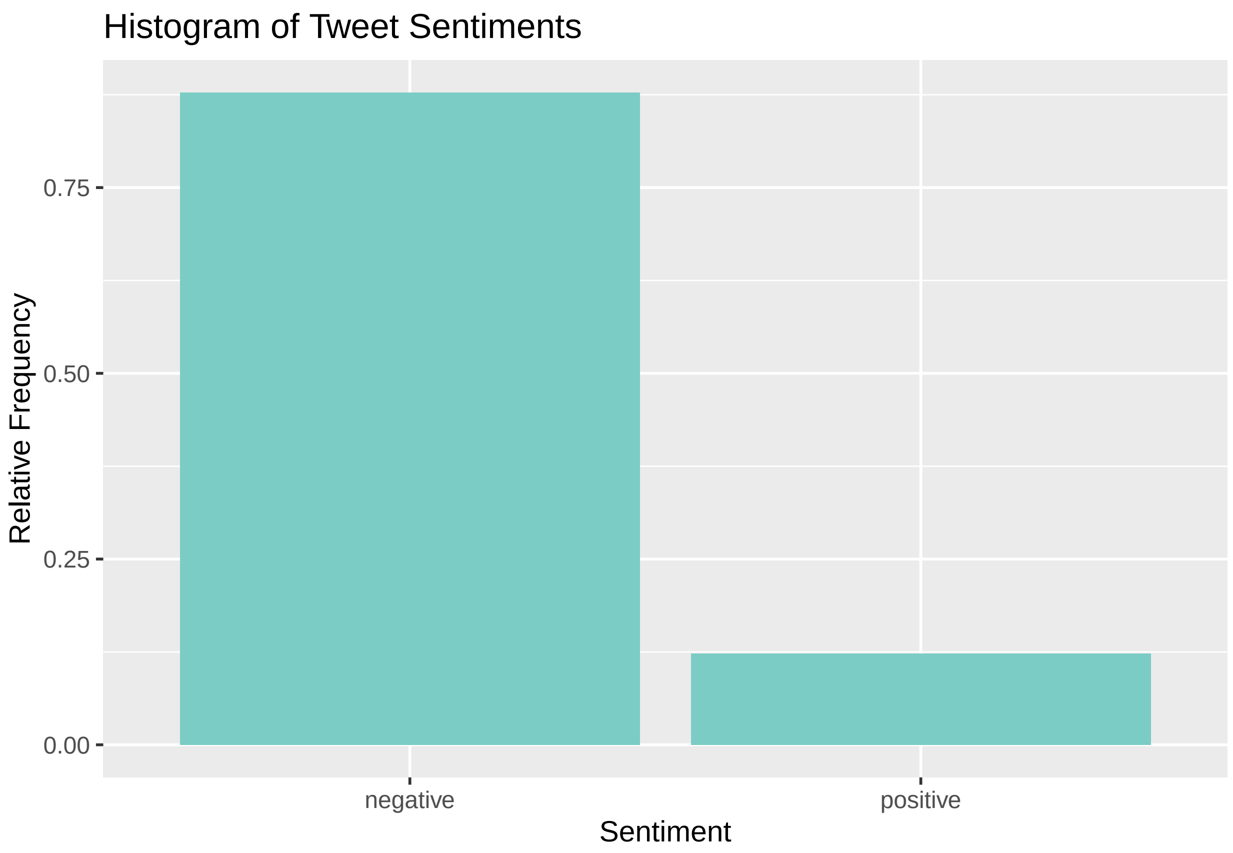
```
sum(bing_summary[bing_summary$sentiment == "negative", ]$n)
```

```
## [1] 2114
```

```
sum(bing_summary$n)
```

```
## [1] 2409
```

```
ggplot(bing_summary, aes(x = sentiment, y = n/sum(n))) +
  geom_bar(stat="identity", fill = "#7bccc4") +
  xlab("Sentiment") +
  ylab("Relative Frequency") +
  ggtitle("Histogram of Tweet Sentiments" )
```


Histogram of Tweet Sentiments
Relative Frequency
0.75
0.50
0.25
0.00
negative
positive
Sentiment