Tweet Text Mining

August 16, 2018

```
library(readxl)
library(tidyverse)
```

```
## -- Attaching packages ----------------------------------------------------------- tidyverse 1.2.1 --
```

```
## v ggplot2 3.0.0     v purrr   0.2.5
## v tibble  1.4.2     v dplyr   0.7.6
## v tidyr   0.8.1     v stringr 1.3.1
## v readr   1.1.1     v forcats 0.3.0
```

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

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following object is masked from 'package:base':
##
##     date
```

```
library(tm)
```

```
## Loading required package: NLP
```

```
##
## Attaching package: 'NLP'
```

```
## The following object is masked from 'package:ggplot2':
##
##     annotate
```

```
library(wordcloud)
```

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

```
library(janitor)

# step 1: read data
freddieGray <- read_excel("C:/Users/panke/Downloads/WFU/R/fianl exam practice/freddieGray.xlsx")
# step 2: eliminate duplicated tweets
fg <- freddieGray %>% distinct(Tweet, .keep_all = TRUE)
```

```
# step 3: write function (use lubridate)
tweetHour <- function(x) {
  hr <- hour(x)
  if (hr < 12) {
    return("Morning")
  } else if (hr < 17) {
    return("Afternoon")
  } else {
    return("Evening")
```

```
  }
}

# test function
tweetHour("2015-04-26 12:50:21")
```

```
## [1] "Afternoon"
```

```
# works well move on to for loop

# step 3: write loop to apply function
# create input with small data to test if this loop works faster
input <- c("2015-04-16 16:26:56", "2015-04-16 07:26:56", "2015-04-16 19:26:56")
output <- vector("character", length(input))

for (i in seq_along(input)) {
  output[[i]] <- tweetHour(input[[i]])
}

# check results
output
```

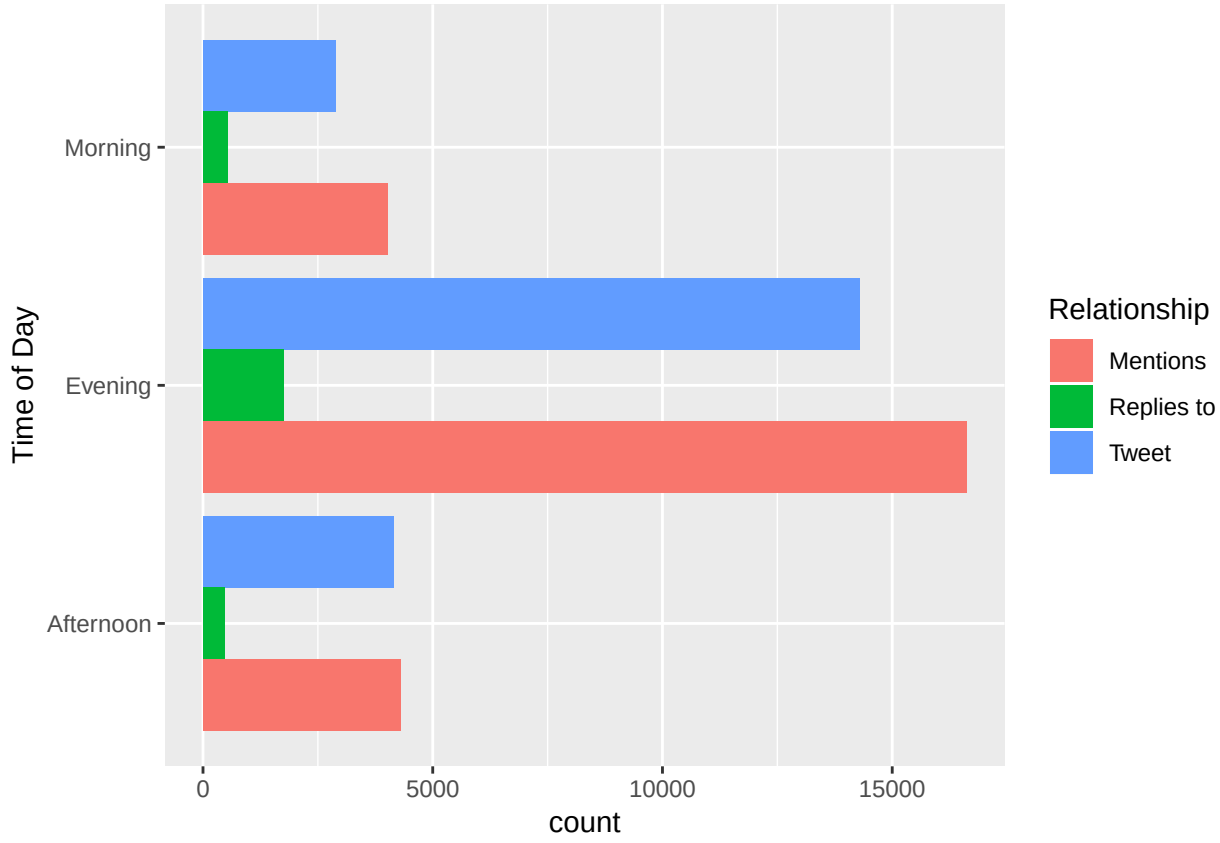
```
## [1] "Afternoon" "Morning"   "Evening"
```

```
# works well

# put loop inside function for efficiency and use the fg data
input <- fg$`Tweet Date (UTC)`
output <- vector("character", length(input))

for (i in seq_along(input)) {

  tweetHour <- function(x) {
    hr <- hour(x)
    if (hr < 12) {
      return("Morning")
    } else if (hr < 17) {
      return("Afternoon")
    } else {
      return("Evening")
    }
  }

  output[[i]] <- tweetHour(input[[i]])
  }

# step 4: write loop to count how many tweets in morning, afternoon, evening
m = 0
a = 0
e = 0

for (i in seq_along(output)) {
   if(output[[i]] == "Morning") {m=m+1}
  else if(output[[i]] == "Afternoon") {a=a+1}
  else{e=e+1}
```

```
}

result <- c(Morning = m, Afternoon = a, Evening = e)
result
```

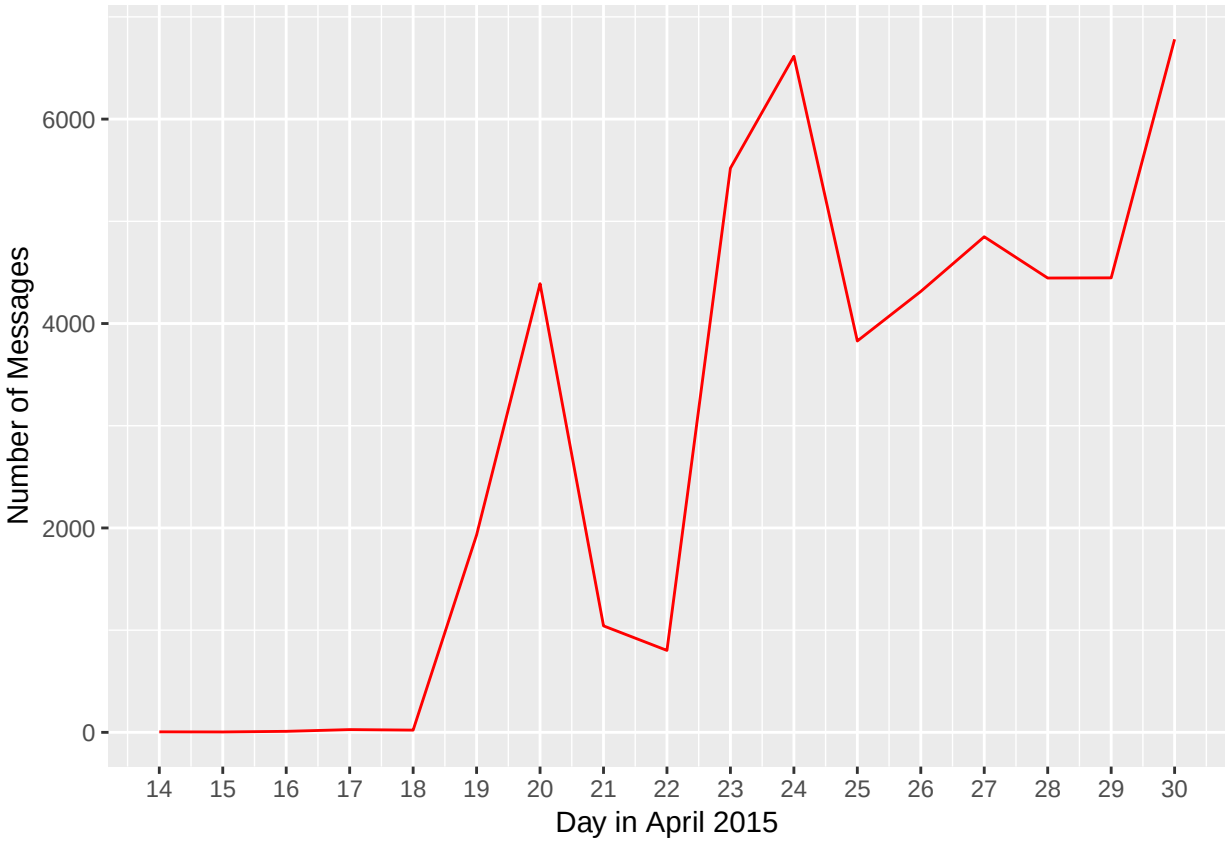
```
##   Morning Afternoon   Evening
##      7446      8914     32673
```

```
# step 5: use transformation & visualization to illustrate differences in number of messages across mes
variables <- c("Vertex 1", "Vertex 2", "Latitude", "Longitude")

fgT <- fg  %>%
  remove_empty(., which = "cols") %>%
  mutate(., `Time of Day` = output) %>%
  select(., one_of(variables),Tweet,`Tweet Date (UTC)`, `Time of Day`,
         Relationship)

ggplot(data = fgT) +
  geom_bar(mapping = aes(x = `Time of Day`, fill = Relationship),
                         position = "dodge") + coord_flip()
```




```
# more mentions and tweets are posted in general, when compared to replies to
# retweets are recorded as mentions, this could explain why we have so many mentions in the data

# check number of messages over time--do we see any day patterns?
fgD <- fgT %>%
  count(day(`Tweet Date (UTC)`))
```

```r
colnames(fgD) <- c("Day in April 2015", "Number of Messages")

ggplot(data = fgD) +
  geom_line(mapping = aes(x = `Day in April 2015`, y = `Number of Messages`), color = "red")+
  scale_x_continuous(name = "Day in April 2015", breaks = c(14:30))
```




```r
# spike in number of messages on days 20, 23, 24, and 30
# Freedie Gray died on the 12 and so protests reach a peak (it seems) about a week after the incident t

# step 6: what words appear the most in messages about death, police, and violence
words <- c("police", "death", "die", "violence")
fgT$Tweet<- str_to_lower(fgT$Tweet)
usableText=str_replace_all(fgT$Tweet,"[^[:graph:]]", " ")

(fgP=fgT %>% filter(str_detect(usableText, "police")))
```

```
## # A tibble: 11,153 x 8
##    `Vertex 1` `Vertex 2` Latitude Longitude Tweet `Tweet Date (UTC)`
##    <chr>      <chr>         <dbl>     <dbl> <chr> <dttm>
##  1 humilitypi theroot          NA        NA #fre~ 2015-04-16 10:26:29
##  2 muniqui19  deray            NA        NA rt @~ 2015-04-16 16:22:25
##  3 oldsilasw~ baltimore~       NA        NA rt @~ 2015-04-16 18:13:09
##  4 itsmikebi~ baltimore~       NA        NA rt @~ 2015-04-17 00:26:05
##  5 frani20    baltimore~       NA        NA "rt ~ 2015-04-17 02:41:44
##  6 notthatra~ katylied67       NA        NA rt @~ 2015-04-17 03:38:49
##  7 patzyjo    citizen__b       NA        NA "rt ~ 2015-04-17 08:10:23
```

```
##  8 katylied67 katylied67       NA        NA the ~ 2015-04-17 03:32:23
##  9 goodhumou~ baltimore~       NA        NA "rt ~ 2015-04-17 10:24:11
## 10 hicksfilo~ baltimore~       NA        NA "rt ~ 2015-04-17 13:29:14
## # ... with 11,143 more rows, and 2 more variables: `Time of Day` <chr>,
## #   Relationship <chr>
```

```
(fgDD=fgT %>% filter(str_detect(fgT$Tweet, "death")))
```

```
## # A tibble: 2,270 x 8
##    `Vertex 1` `Vertex 2` Latitude Longitude Tweet `Tweet Date (UTC)`
##    <chr>      <chr>         <dbl>     <dbl> <chr> <dttm>
##  1 votenanoc~ baltimore~       NA        NA @bal~ 2015-04-19 14:37:44
##  2 nicky2thi~ baltimore~       NA        NA rt @~ 2015-04-19 14:41:45
##  3 slickrick~ uncle_qui~       NA        NA rt @~ 2015-04-19 14:44:33
##  4 auntieimp~ auntieimp~       NA        NA poli~ 2015-04-19 15:08:57
##  5 cosmicife  mattbutle~       NA        NA rt @~ 2015-04-19 15:13:22
##  6 darlingne~ seanjjord~       NA        NA rt @~ 2015-04-19 16:28:05
##  7 oneofakin~ seabethree       NA        NA rt @~ 2015-04-19 16:42:12
##  8 missjones~ mayorsrb         NA        NA rt @~ 2015-04-19 16:43:29
##  9 no_cut_ca~ passthami~       NA        NA "rt ~ 2015-04-19 16:44:53
## 10 noelieulu~ longhouse~        0         0 @kas~ 2015-04-19 16:34:39
## # ... with 2,260 more rows, and 2 more variables: `Time of Day` <chr>,
## #   Relationship <chr>
```

```
fgWC <- rbind(fgP, fgDD)

treat_corpus <- Corpus(VectorSource(fgWC$Tweet))
treat_corpus <- tm_map(treat_corpus, removePunctuation)
```

```
## Warning in tm_map.SimpleCorpus(treat_corpus, removePunctuation):
## transformation drops documents
```

```
treat_corpus <- tm_map(treat_corpus, removeNumbers)
```

```
## Warning in tm_map.SimpleCorpus(treat_corpus, removeNumbers): transformation
## drops documents
```

```
treat_corpus <- tm_map(treat_corpus, stripWhitespace)
```

```
## Warning in tm_map.SimpleCorpus(treat_corpus, stripWhitespace):
## transformation drops documents
```

```
treat_corpus <- tm_map(treat_corpus, removeWords, stopwords("english"))
```

```
## Warning in tm_map.SimpleCorpus(treat_corpus, removeWords,
## stopwords("english")): transformation drops documents
```

```
treat_corpus <- tm_map(treat_corpus, removeWords, c("freddiegray", "police", "death"))
```

```
## Warning in tm_map.SimpleCorpus(treat_corpus, removeWords,
## c("freddiegray", : transformation drops documents
```

```
#tm_map(treat_corpus, function(x) iconv(enc2utf8(x), sub = "byte"))
#tm_map(treat_corpus, function(x) iconv(x, to='UTF-8-MAC', sub='byte'))
#wordcloud(treat_corpus, max.words=100, min.freq=5, random.order = F, colors=brewer.pal(8, "Dark2"))
```