Project 3 : Sentiment Analysis of tweets related to Siri

Aparna Cleetus (axc190011), Obuli Vignesh Rangasamy (oxr170630)

2020-07-17

```
# Import all the libraries
library(rtweet)
```

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

```
library(sentimentr)
```

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

```
library(ndjson)
```

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

```
##
## Attaching package: 'ndjson'
```

```
## The following object is masked from 'package:rtweet':
##
##     flatten
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.6.3
```

```
library(tidyverse)
```

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

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

```
## v tibble  3.0.1     v dplyr   0.8.3
## v tidyr   1.1.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
## v purrr   0.3.4
```

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

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

```
## Warning: package 'readr' was built under R version 3.6.3
## Warning: package ’purrr’ was built under R version 3.6.3
```

```
## Warning: package ’forcats’ was built under R version 3.6.3
```

```
## -- Conflicts ----------------------------------------------------------------- tidyverse_conflicts()
## x dplyr::filter()  masks stats::filter()
## x purrr::flatten() masks ndjson::flatten(), rtweet::flatten()
## x dplyr::lag()     masks stats::lag()
```

```
library(tidytext)
```

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

```
# Get Siri related tweets using search_stream()
siri_tweets <-search_tweets(q="#siri", n =10000, include_rts =FALSE, lang="en")
siri_tweets$stripped_text <- gsub("http.*","",  siri_tweets$text)
siri_tweets$stripped_text <- gsub("https.*","", siri_tweets$stripped_text)
siri_tweets$stripped_text <- gsub("#*","", siri_tweets$stripped_text)
siri_tweets$stripped_text <- gsub("@*","", siri_tweets$stripped_text)

# Get the approximate sentiment (polarity) of each tweet
siri_sentiment <-sentiment_by(siri_tweets$stripped_text)
head(siri_sentiment)
```

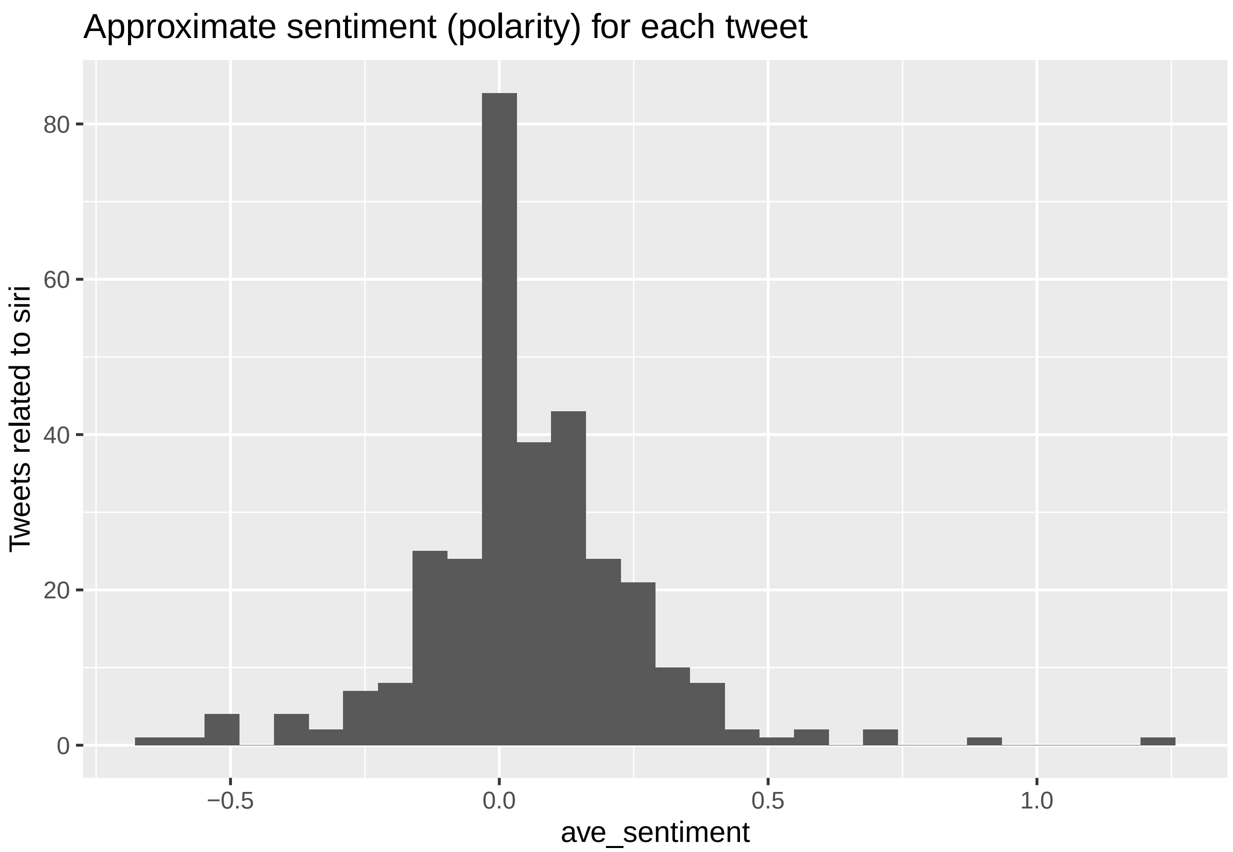
```
##    element_id word_count         sd ave_sentiment
## 1:          1         36 0.08401681   -0.07105026
## 2:          2         55 0.42260490   -0.04062790
## 3:          3         39 0.49746295   -0.10175943
## 4:          4         41 0.48113832   -0.07295493
## 5:          5         45 0.42358994   -0.04132443
## 6:          6         40 0.48619170   -0.08559056
```

```
summary(siri_sentiment$ave_sentiment)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.62554 -0.01880  0.02391  0.05475  0.15912  1.24456
```

```
# Plot the approximate sentiment (polarity) for each tweet
ggplot(siri_sentiment, aes(x=ave_sentiment)) +
      geom_histogram() +
      labs(x ="ave_sentiment",y ="Tweets related to siri", title = "Approximate sentiment (polarity) fo
```

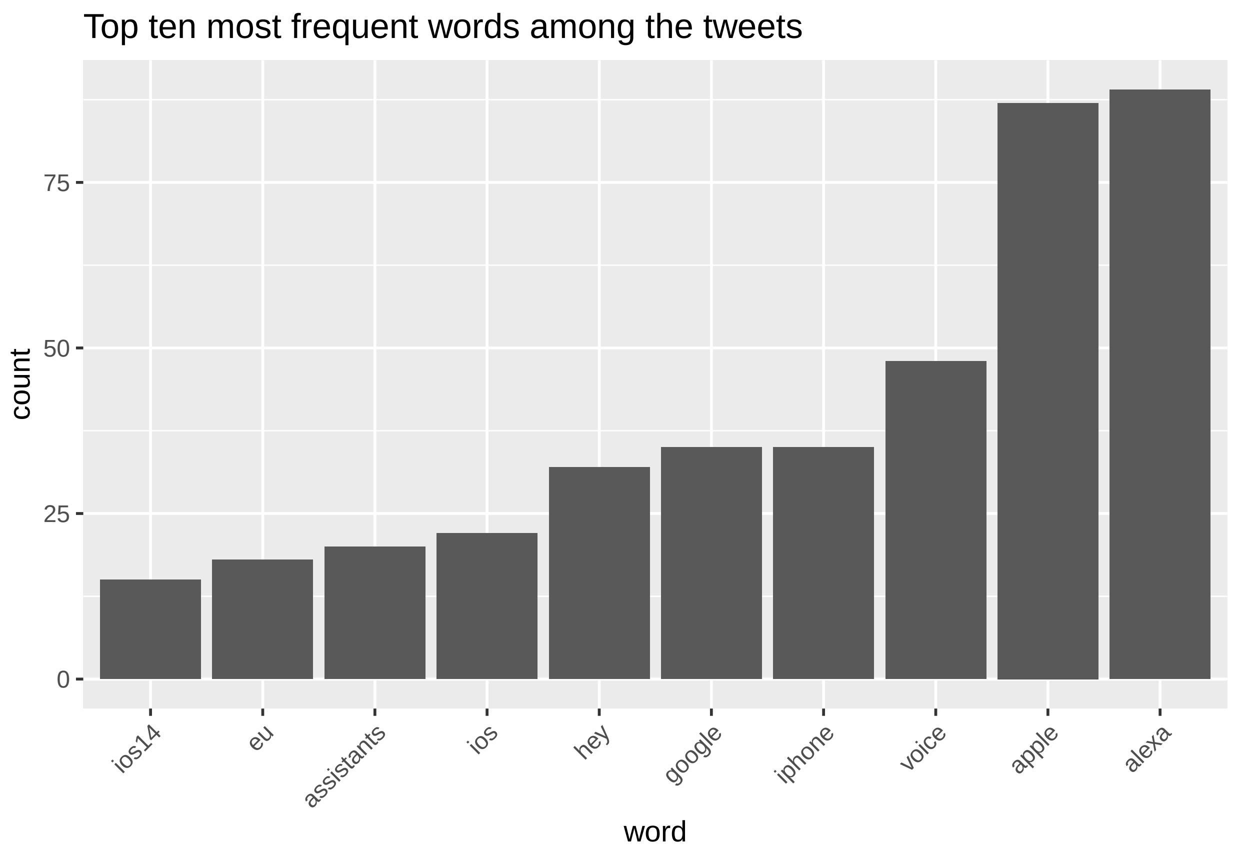
```
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
```



```
# Siri tweets analysis
# Get the most frequent words
siri_tokens <-siri_tweets %>%
      select(text) %>%
      unnest_tokens(word, text)

#plot the top ten frequent words in the tweets
siri_tokens %>%
  group_by(word) %>%
  summarise(count =n()) %>%
  anti_join(stop_words) %>%
  filter(!word %in%c('https', 't.co', 'siri99', 'siri', 'amp', 'gt')) %>%
  arrange(desc(count)) %>%top_n(10) %>%
  mutate(word =reorder(word, count)) %>%
  ggplot(aes(x =word, y =count)) +geom_col() +
  theme(axis.text.x =element_text(angle=45, hjust=1)) +
  labs(title ="Top ten most frequent words among the tweets")
```

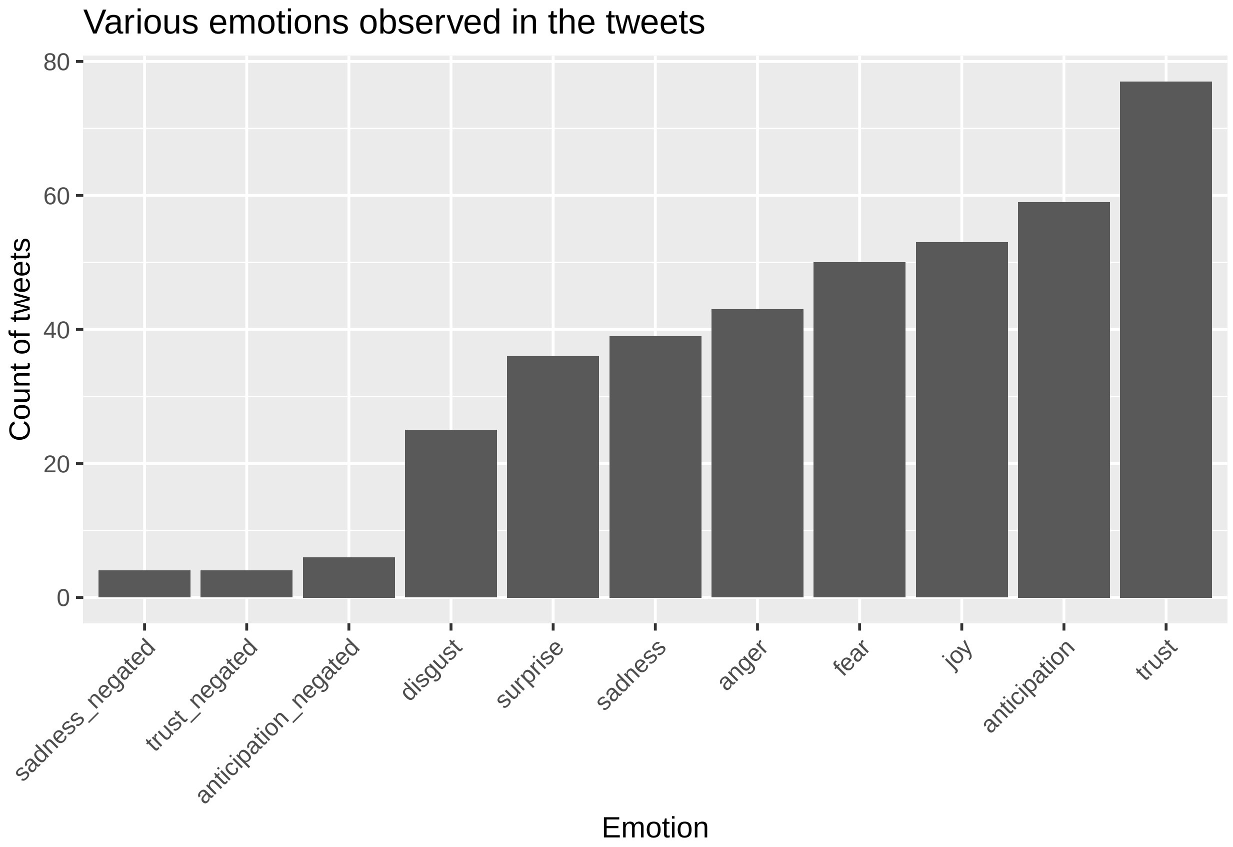
```
## Joining, by = "word"
```

```
## Selecting by count
```



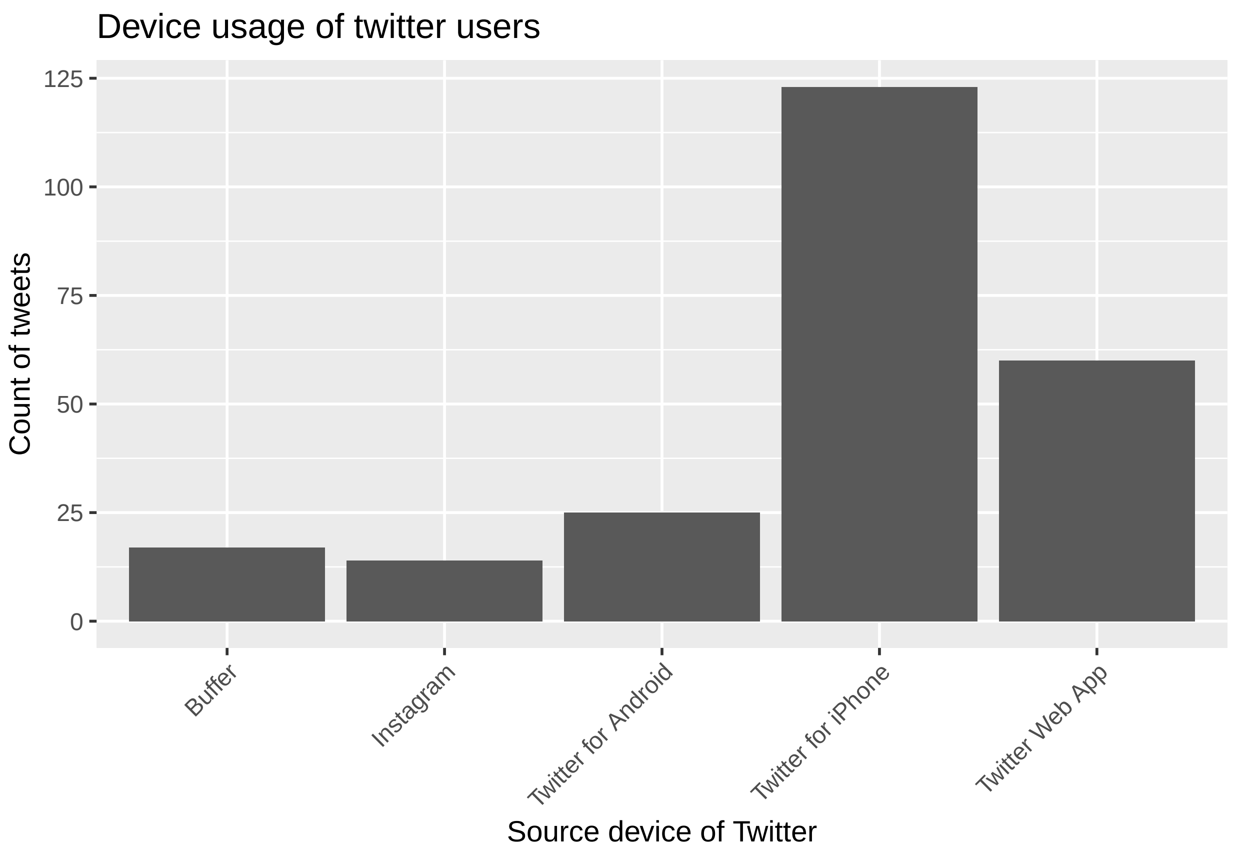
```
# Understand Siri tweets emotions
siri_emotions <-emotion_by(get_sentences(siri_tweets$stripped_text))

siri_emotions <-siri_emotions %>%
      group_by(element_id) %>%
      filter(emotion_count ==1)

#plot of the emotions observed for the tweets
siri_emotions %>%
  group_by(emotion_type) %>%
  summarise(count =n()) %>%
  top_n(10) %>%
  mutate(emotion_type =reorder(emotion_type, count)) %>%
  ggplot(aes(x=emotion_type, y =count)) +
  geom_col() +theme(axis.text.x =element_text(angle=45, hjust=1)) +
  labs(x="Emotion", y= "Count of tweets",title ="Various emotions observed in the tweets")
```

```
## Selecting by count
```

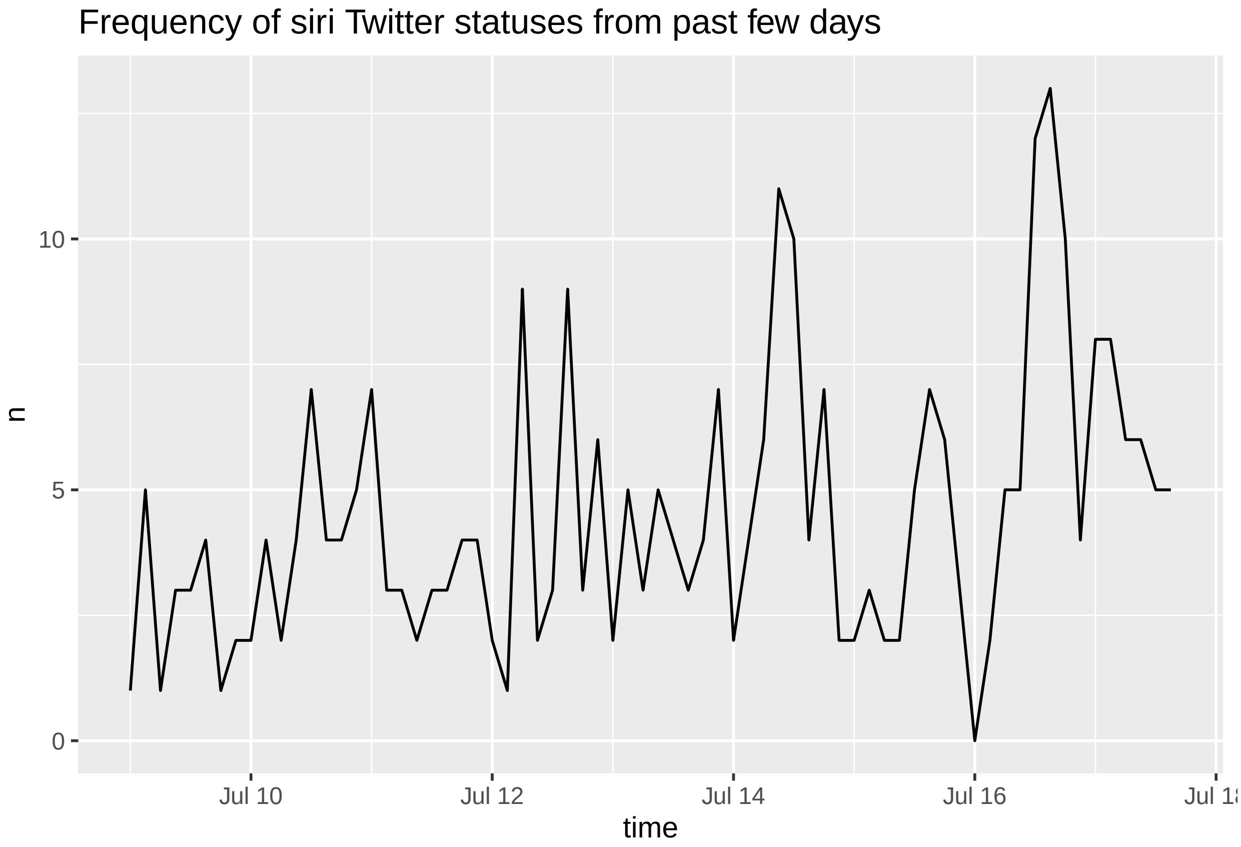


```
# Generic statistics about the tweets
siri_tweets %>%
  group_by(source) %>%
  summarise(count =n()) %>%
  top_n(5) %>%arrange(desc(count)) %>%
  ggplot(aes(x=source, y =count)) +
  geom_col() +theme(axis.text.x =element_text(angle=45, hjust=1)) +
  labs(y ="Count of tweets",x ="Source device of Twitter",title ="Device usage of twitter users")
```

```
## Selecting by count
```



```
# plot time series of tweets
siri_tweets %>%
  ts_plot("3 hours") +
  labs(title ="Frequency of siri Twitter statuses from past few days")
```


Frequency of siri Twitter statuses from past few days
n
10
5
0
Jul 10
Jul 12
Jul 14
Jul 16
Jul 18
time