

SentimentAnalysis

Project – 2

Ashika Prakash Acharya (axa190084)

2020-04-09

```
#####  
# Import all the libraries  
#####  
  
library(rtweet)  
library(sentimentr)  
library(ndjson)  
  
##  
## Attaching package: 'ndjson'  
  
## The following object is masked from 'package:rtweet':  
##  
##   flatten  
  
library(ggplot2)  
library(tidyverse)  
  
## — Attaching packages  


---

  
tidyverse 1.3.0 —  
  
## ✓ tibble 3.0.0      ✓ dplyr 0.8.5  
## ✓ tidyr 1.0.2       ✓ stringr 1.4.0  
## ✓ readr 1.3.1       ✓ forcats 0.5.0  
## ✓ purrr 0.3.3  
  
## Warning: package 'tibble' was built under R version 3.6.2  
  
## — 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)
```

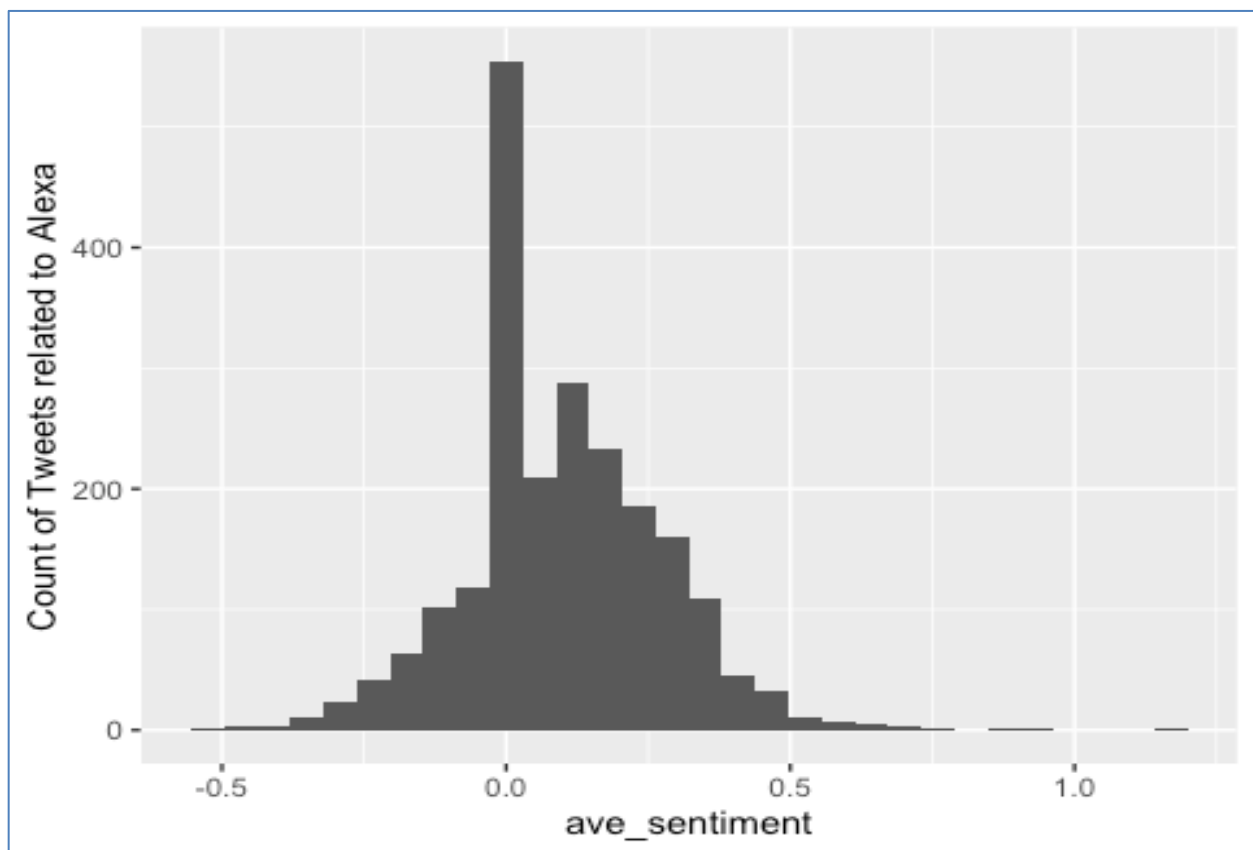
Sentiment Analysis using package 'sentimentr'

```
#####  
# Get ALEXA related tweets using search_stream()  
#####  
  
alexa_tweets <- search_tweets(q="#alexa", n = 10000, include_rts = FALSE,  
lang="en")  
  
#####  
# Sentiment Extraction  
#####  
  
alexa_sentiment <- sentiment_by(alexa_tweets$text)  
  
## Warning: Each time `sentiment_by` is run it has to do sentence boundary  
disambiguation when a  
## raw `character` vector is passed to `text.var`. This may be costly of time  
and  
## memory. It is highly recommended that the user first runs the raw  
`character`  
## vector through the `get_sentences` function.  
  
head(alexa_sentiment)  
  
##   element_id word_count      sd ave_sentiment  
## 1:         1         51 0.1740237   -0.03625007  
## 2:         2         18 0.1767767   -0.14949425  
## 3:         3         16 0.0000000    0.00000000  
## 4:         4         15      NA    0.00000000  
## 5:         5         14      NA    0.00000000  
## 6:         6         14      NA    0.21380899  
  
summary(alexa_sentiment$ave_sentiment)  
  
##      Min.   1st Qu.   Median     Mean   3rd Qu.     Max.  
## -0.54569  0.00000   0.08175  0.09621  0.20750  1.14945
```

```
#####
# Sentiment Plots
#####

ggplot(alexa_sentiment, aes(x=ave_sentiment)) +
  geom_histogram() +
  labs(x = "ave_sentiment",
       y = "Count of Tweets related to Alexa")

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



Observation: We can see most of the sentiments are either neutral or towards positive and few of them are on the negative side.

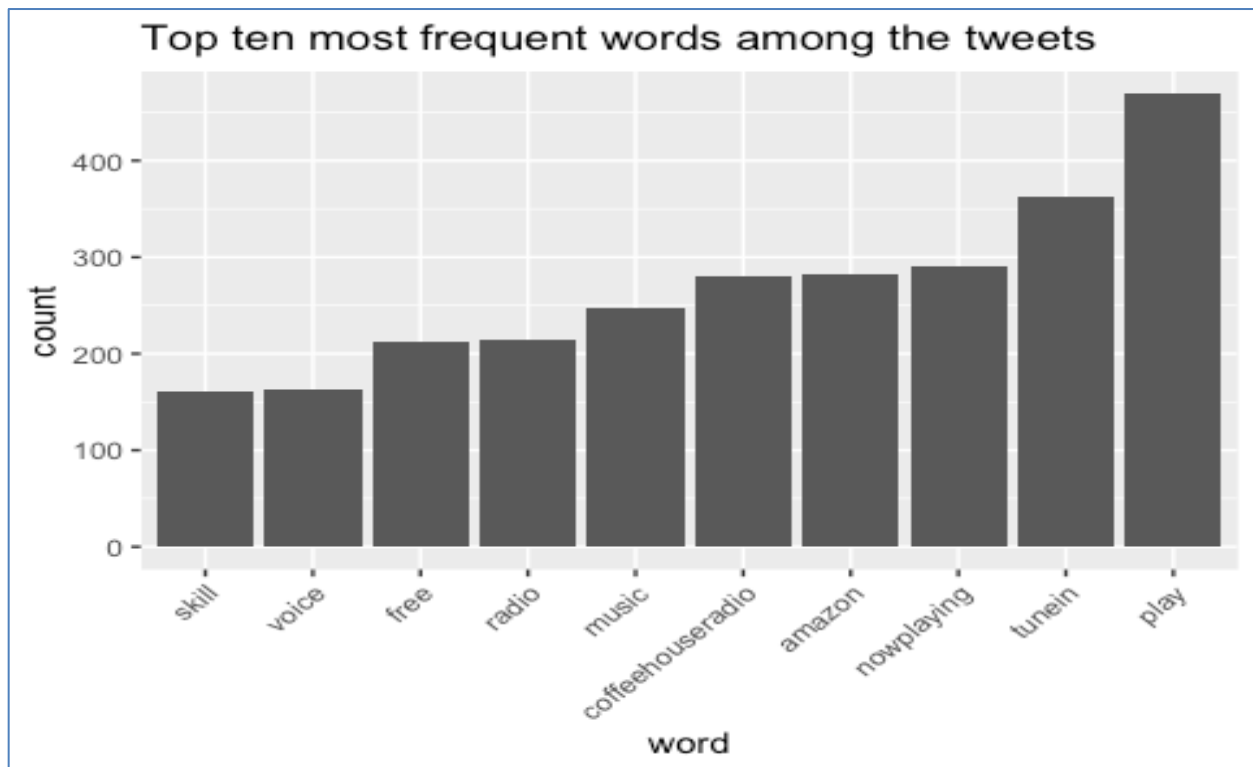
-1 → negative

+1 → positive

```
#####
# Alexa tweets analysis
#####
# Most frequent words
#####

alex_tokens <- alexa_tweets %>%
  select(text) %>%
  unnest_tokens(word, text)

alex_tokens %>%
  group_by(word) %>%
  summarise(count = n()) %>%
  anti_join(stop_words) %>%
  filter(!word %in% c('https', 't.co', 'alexa99', 'alexa', '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")
```

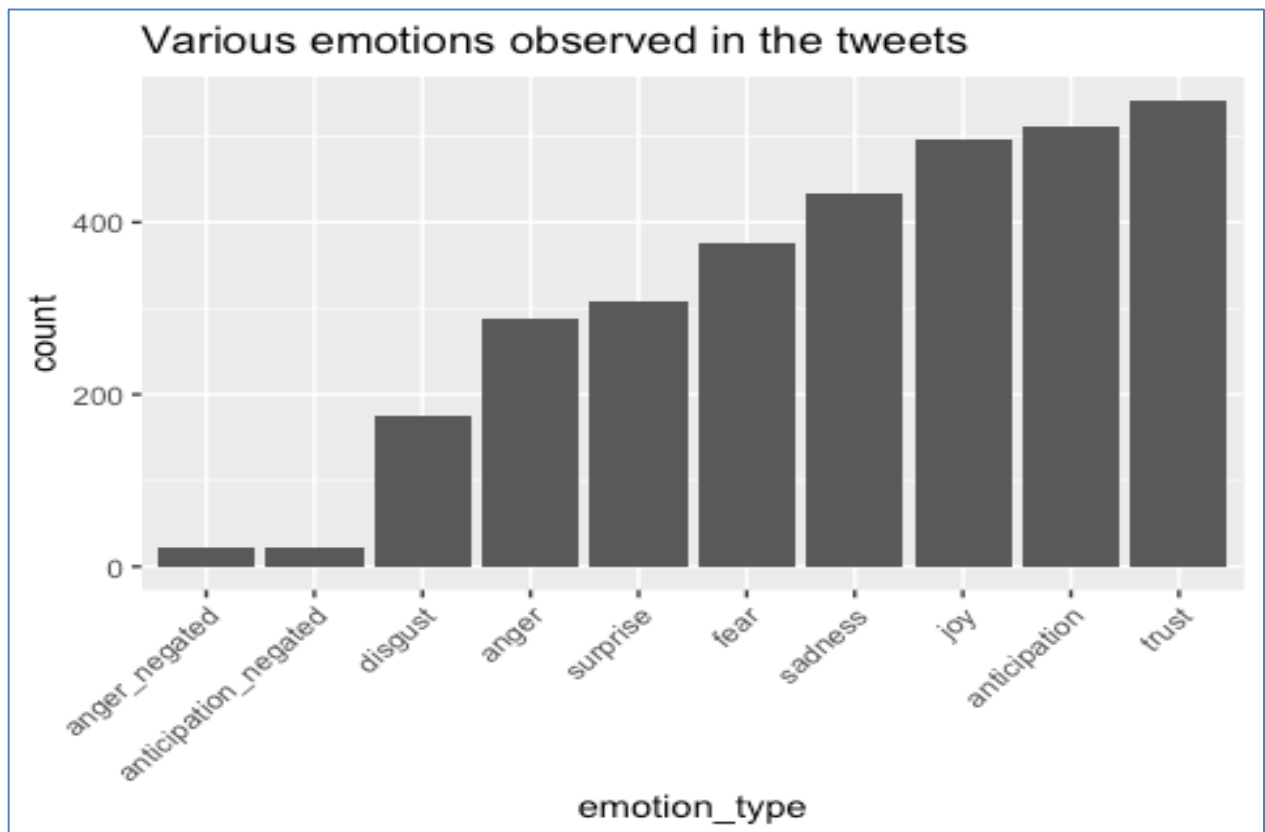


Top 10 words that are used in the tweets related to Alexa is shown above.

Understand Alexa tweets emotions

```
alex_a_emotions <- emotion_by(get_sentences(alex_a_tweets$text))
```

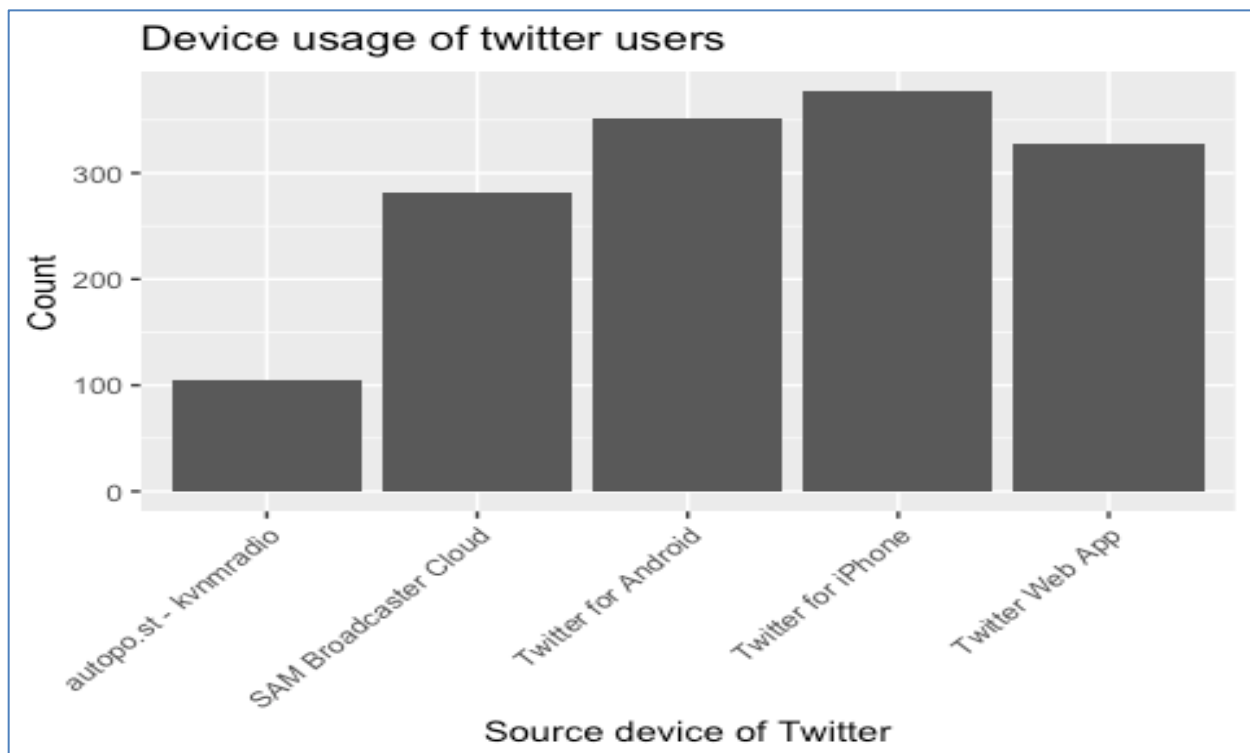
```
alex_a_emotions <- alex_a_emotions %>%  
  group_by(element_id) %>%  
  filter(emotion_count == 1)  
  
alex_a_emotions %>%  
  group_by(emotion_type) %>%  
  summarise(count = n()) %>%  
  top_n(10) %>%  
  mutate(emotion_type = reorder(emotion_type, count)) %>%  
  ggplot(aes(x=emotion_type, y = count)) +  
  geom_col() +  
  theme(axis.text.x = element_text(angle=45, hjust=1)) +  
  labs(title = "Various emotions observed in the tweets")
```



We have used **emotion_by()** function that is available with 'sentimentr' package and assigns an emotion to each tweet.

A plot showing top 10 emotions to which the tweets belong. We can see a mixture of positive and negative emotions that the tweets show.

```
#####  
# Generic statistics about the tweets  
#####  
  
alex_a_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",  
       x = "Source device of Twitter",  
       title = "Device usage of twitter users")
```



Twitter users who posted about Alexa vary almost equally among Iphone and Android.

```
## plot time series of tweets
```

```
alex_tweets %>%  
  ts_plot("3 hours") +  
  labs(title = "Frequency of #alex Twitter statuses from past few days")
```

