Sentimental Analysis on Reviews of Amazon Product

Opeyemi Alabi

This is the CONTINUATION of the **Amazon Customer Review Analysis** done with Python.

```
#Importing relevant libraries
library(dplyr)
library(rvest)
library(stringr)
library(tm)
library(wordcloud2)
library(ggplot2)
library(tidytext)
```

```
#Read csv -- and print shape of the data.
df <- read.csv('Cleaned_review_output.csv')
print(dim(df))
```

```
## [1] 14337     3
```

```
#Tokenizing the customer review column and setting the "ngram" to 2
tokens <- df %>%
  unnest_tokens(words, c(Cleaned_Review), token = 'ngrams', n = 2)
```

```
#Selecting only the relevant columns needed
tokens <- tokens %>%
  select(c('Product','ReviewStar','words'))

print(head(tokens,4))
```

```
##            Product ReviewStar        words
## 1 boAt Rockerz 255    Neutral  doubt great
## 2 boAt Rockerz 255    Neutral   great bass
## 3 boAt Rockerz 255    Neutral   bass great
## 4 boAt Rockerz 255    Neutral great extent
```

After tokenizing the data… the shape of the data has been changed **-** the data now consist of **170306 rows** compared to **14337** earlier.

```
#Checking the structure of the data
str(tokens)
```

```
## 'data.frame':    170306 obs. of  3 variables:
##  $ Product   : chr  "boAt Rockerz 255" "boAt Rockerz 255" "boAt Rockerz 255" "boAt Rockerz 25
5" ...
##  $ ReviewStar: chr  "Neutral" "Neutral" "Neutral" "Neutral" ...
##  $ words     : chr  "doubt great" "great bass" "bass great" "great extent" ...
```

```
# Converting columns to "Factor" data type
tokens$words <- as.factor(tokens$words)
tokens$Product <- as.factor(tokens$Product)
tokens$ReviewStar <- as.factor(tokens$ReviewStar)

print(str(tokens))
```

```
## 'data.frame':    170306 obs. of  3 variables:
##  $ Product   : Factor w/ 10 levels "boAt Rockerz 255",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ ReviewStar: Factor w/ 3 levels "Negative","Neutral",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ words     : Factor w/ 90162 levels "_charger cable",..: 22279 34312 6263 34414 28004 52957
12743 19765 73951 14761 ...
## NULL
```

```
#Checking the order of the product sales to know which product sold most..
print(sort(table(tokens$Product),decreasing = T))
```

```
## 
##      boAt Rockerz 255   Sennheiser CX 6.0BT            JBL T110BT 
##                 66578                 61217                 16132 
##            JBL T205BT         PTron Intunes Skullcandy S2PGHW-174 
##                 14439                  3481                  2992 
## Samsung EO-BG950CBEIN           Flybot Wave           Flybot Boom 
##                  2476                  1838                   858 
##           Flybot Beat 
##                   295
```

```
#Checking the Review that has the highest rating.
print(sort(table(tokens$ReviewStar), decreasing = T))
```

```
## 
## Positive Negative  Neutral 
##   107108    42421    20777
```

Checking the Frequency of the tokenize customer reviews - this is to know the number of time each word appears in the data.

```
# Printing the first 6 rows from the Frequency.
word_count = as.data.frame(sort(table(tokens$words),
                                decreasing = T))
colnames(word_count) <- c('Word','Frequency')
print(head(word_count))
```

```
##                  Word Frequency
## 1      sound quality      3192
## 2       quality good       927
## 3       battery life       776
## 4 noise cancellation       639
## 5       good product       628
## 6         good sound       618
```

```
#Printing the last 6 rows from the Frequency
print(tail(word_count))
```

```
##                      Word Frequency
## 90157      zero.one star         1
## 90158   zero.please dont         1
## 90159 zero.this earphone         1
## 90160       zindagi main         1
## 90161      zipped pocket         1
## 90162           zl trash         1
```

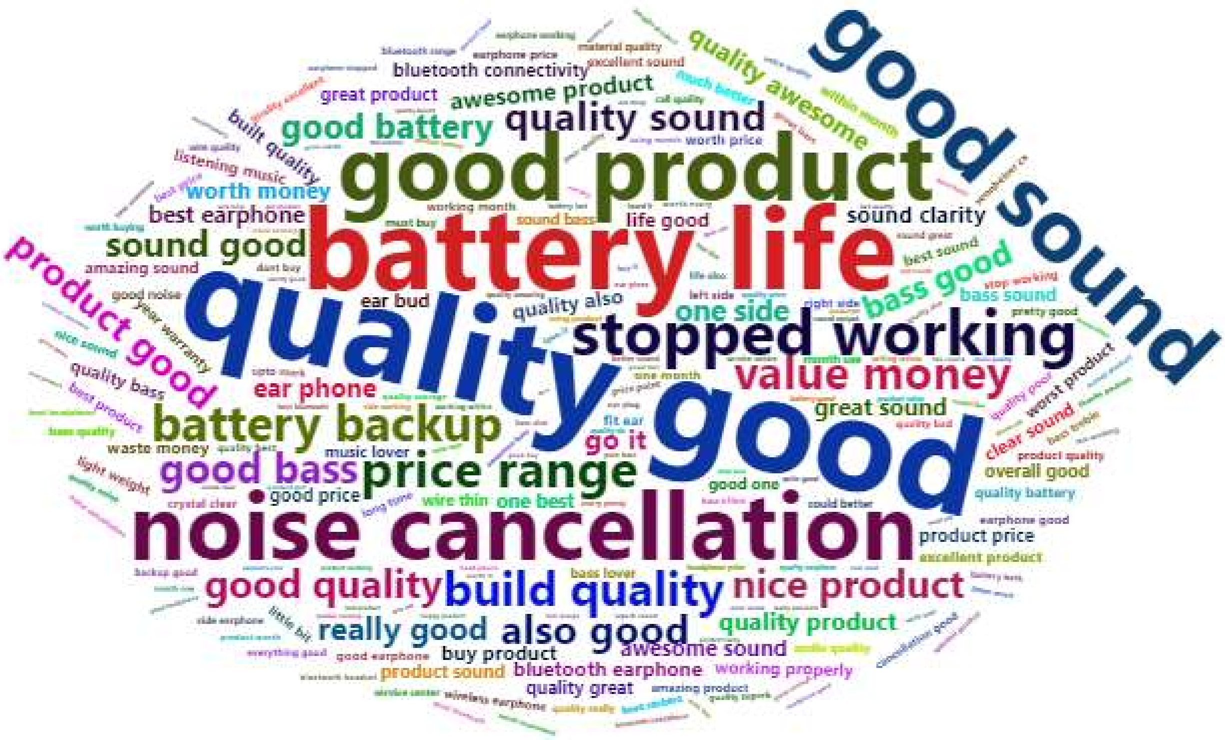
Preparing for **WordCloud** GRAPH PLOT …

```
# Filtering any frequency that is LESS that 40 ...
word_filter <- word_count %>%
  filter(Frequency > 39)

print(tail(word_filter))
```

```
##                 Word Frequency
## 219   sound battery        41
## 220       wire long        41
## 221    love product        40
## 222     product jbl        40
## 223  quality decent        40
## 224 speaker working        40
```

```
#WordCloud Plot
wordcloud2(word_filter, size = 1.3)
```



From the above WordCloud plot… It shows that the **customer comment** falls mostly on **(quality, battery, noise, sound, bass, earphones)** concerning the products.

Also, it's necessary to identify the **specific product** that the **comment** is relating to, and also identify the Kind of **Review** that the customer make on that product.

Therefore am going to use the **words** in the **WordCloud** plot to perform an analysis on the products by plotting a graph that will **map** each words to its **corresponding products** and its **corresponding reviews**.

```
#Asigning the words to a variable - "Keywords"
keywords <- as.character(word_filter$Word)
print(head(keywords))
```

```
## [1] "sound quality"      "quality good"       "battery life"
## [4] "noise cancellation" "good product"       "good sound"
```

```
#Filtering those Keywords in the dataframe and counting the
#number of its occurrence.
output <- tokens %>%
  filter(words %in% c(keywords)) %>%
  count(Product, ReviewStar, words, sort = T)
print(head(output))
```

```
##                Product ReviewStar         words   n
## 1    boAt Rockerz 255   Positive sound quality 947
## 2 Sennheiser CX 6.0BT   Positive sound quality 880
## 3    boAt Rockerz 255   Positive  battery life 399
## 4    boAt Rockerz 255   Positive  quality good 276
## 5    boAt Rockerz 255   Positive  good product 228
## 6 Sennheiser CX 6.0BT   Positive  quality good 217
```

```
# Rechecking the shape of the data.
print(dim(output))
```

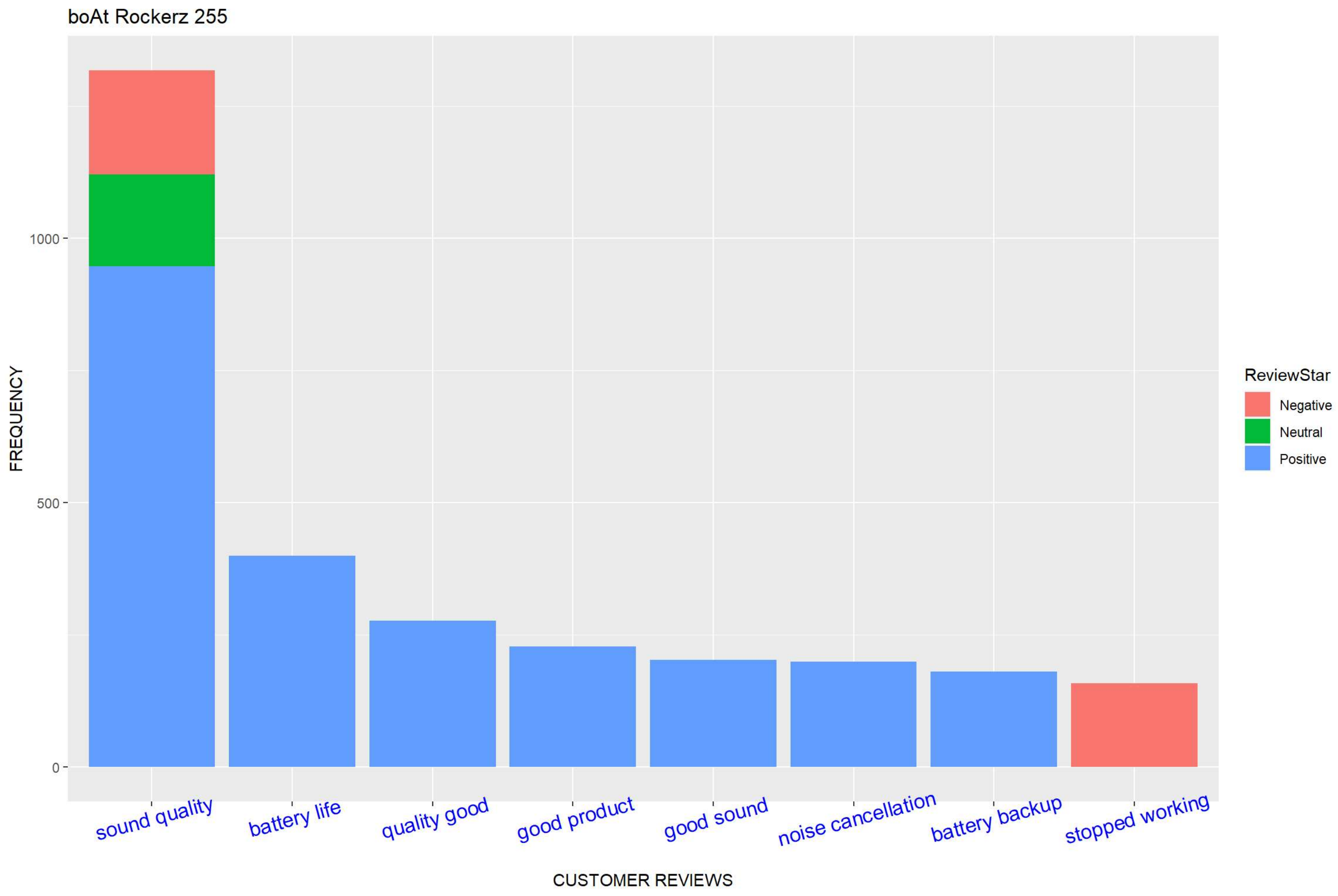
```
## [1] 2821    4
```

GRAPH PLOT

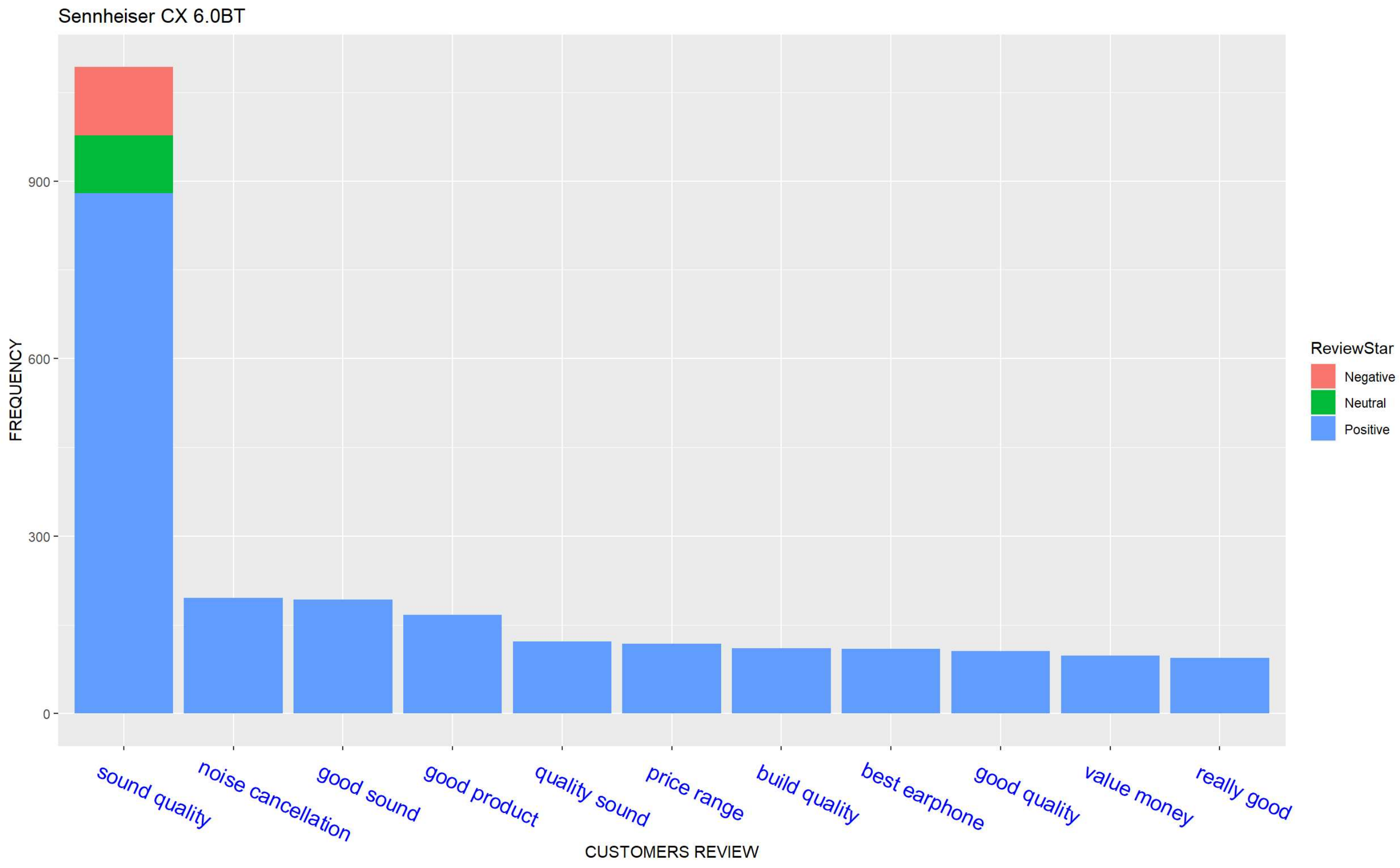
The graphical plot analysis will only be performed on the two products that has the highest sales which are:

1. boAt Rockerz 255
2. Sennheiser CX 6.0BT

```
# ggplot graph for First Product
output %>%
  filter(Product == "boAt Rockerz 255" & n > 150) %>%
  ggplot(aes(x = reorder(words, -n), y = n)) +
  geom_col(aes(fill = ReviewStar)) +
  labs(y = 'FREQUENCY', x = 'CUSTOMER REVIEWS',
       title ="boAt Rockerz 255") +
  theme(axis.text.x = element_text(angle = 15, colour = 'blue',
                                   size = 13))
```



```
# ggplot graph for Second Product
 output %>%
  subset(words!='quality good') %>%
  filter(Product == 'Sennheiser CX 6.0BT' & n > 90) %>%
  ggplot(aes(x = reorder(words,-n), y = n)) +
  geom_col(aes(fill = ReviewStar)) +
  labs(x = 'CUSTOMERS REVIEW', y = 'FREQUENCY',
       title = 'Sennheiser CX 6.0BT') +
  theme(axis.text.x = element_text(angle = -25,
                                   color = 'blue',size = 14, hjust = 0.2))
```



Summary

1. From the ggplot graph, **boAt Rockerz 255** has much positive review text because most of the **Users** comment very well in terms of the (*sound, noise, battery, quality*) while only few **Users** express their dissatisfaction concerning the *quality* and also how it *stopped working*.

2. Likewise also the user of **Sennheiser CX 6.0BT** has a much positive review about the product. Only a few of the users were not satisfied in terms of the *sound quality*.