Amazon Reviews NLP

Guillermo Carrera Trasobares

1/28/2022

Repository: https://github.com/gcarrerat/NLP_MUII

Introduction

The objective of this document is to realize text mining operations with R. The main goal is to use sentiment analysis to accurately read the positivity or negativity of product reviews using AFINN, plotting them and analyzing the results.

Aside from pure sentiment analysis, other things have been tested such as finding the most common positive or negative words associated with each sentiment in order to generate plots and display them as word clouds.

The dataset used is from Stanford university: http://snap.stanford.edu/data/amazon/productGraph/categoryFiles/ and contains product reviews and metadata from Amazon, including 143.7 million reviews spanning May 1996 - July 2014.

As the dataset is immense, only a small part containing reviews from cell phones and accessories will be used: http://snap.stanford.edu/data/amazon/productGraph/categoryFiles/reviews_Cell_Phones_and_Accessories_5.json.gz

Sources

The most helpful article was on Yelp review analysis:

Does sentiment analysis work? A tidy analysis of Yelp reviews: http://varianceexplained.org/r/yelp-sentiment/

Other sources used for this project:

- Following Up On "Does Sentiment Analysis Work? A tidy Analysis of Yelp Reviews": http://rstudio-pubs-static.s3.amazonaws.com/306818_2a98f9dc58fd409ba2a9fe0916691103.html
- Sentiment Analysis basics: https://bookdown.org/psonkin18/berkshire/sentiment.html
- Assignment 2 – Movie reviews: https://rstudio-pubs-static.s3.amazonaws.com/466037_c10e13e8392640c6b26ee9092d13c575.html
- Top 10 R Packages For Natural Language Processing (NLP): https://analyticsindiamag.com/top-10-r-packages-for-natural-language-processing-nlp/

Objectives

The main objectives of this exercise are:

- Afinn analysis to check if word sentiment is related to review star rating

- Identify and analyze the most common words, words associated with sentiments, plot them and their associated wordmaps
- Outlier analysis to find reviews that have good sentiment and few stars
- TF-IDF, find how important words are in the collection of reviews

Execution

Remove Objects from environment

```
rm(list = ls())
```

Clean terminal output

```
cat("\014")
```

Import libraries

```
library(dplyr)
library(readr)
library(stringr)
library(jsonlite)
library(kableExtra)
library(textdata)
library(tidytext)
library(data.table)
library(ggplot2)
library(R.utils)
library(xtable)
library(wordcloud)
library(reshape2)
```

Set working directory

```
setwd("~/Projects/NLP_MUII/NLP_Project/")
```

Delete file

```
file.remove("reviews_CellPhones_Accessories.json")
```

```
## [1] TRUE
```

Download and unzip file

```
fileloc <- "reviews_CellPhones_Accessories.json.gz"
download.file("http://snap.stanford.edu/data/amazon/productGraph/categoryFiles/reviews_Cell_Phones_and_
gunzip(fileloc)
```

Read into R

```
cell_reviews_file <- "reviews_CellPhones_Accessories.json"
```

Read_lines creates character vector for each line

```
review_lines <- read_lines(cell_reviews_file, progress = TRUE)
```

```
## indexing reviews_CellPhones_Accessories.json [==========---] 4.24TB/s, eta:  0sindexing reviews_Cell
```

```
reviews_combined <- str_c("[", str_c(review_lines, collapse = ", "), "]")
```

Flatten and turn into a tibble

```
df_reviews <- fromJSON(reviews_combined) %>%
  flatten() %>%
  tbl_df()
```

```
## Warning: `tbl_df()` was deprecated in dplyr 1.0.0.
## Please use `tibble::as_tibble()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
```

Verify that the tibble format works (5 rows only)

```
kable(df_reviews[1:5,]) %>%
  kable_styling("striped", full_width = F, latex_options = "HOLD_position")
```

reviewerID	asin	reviewerName	helpful	reviewText
A30TL5EWN6DFXT	120401325X	christina	0, 0	They look good and stick good! I just don't like the r
ASY55RVNIL0UD	120401325X	emily l.	0, 0	These stickers work like the review says they do. The
A2TMXE2AFO7ONB	120401325X	Erica	0, 0	These are awesome and make my phone look so stylis
AWJ0WZQYMYFQ4	120401325X	JM	4, 4	Item arrived in great time and was in perfect conditio
ATX7CZYFXI1KW	120401325X	patrice m rogoza	2, 3	awesome! stays on, and looks great. can be used on r

Tidy Text

Tidy principles dictate that in a data set, each variable is a column, each observation a row, and each manner of observational unit a table.

Create a unique identifier for each review

```
df_reviews <- df_reviews %>% mutate(reviewID = row_number())
```

Unnest text column

```
df_reviews_words <- df_reviews %>%
  select(asin,reviewID,reviewText,overall) %>%
  unnest_tokens(word,reviewText) %>%
  filter(!word %in% stop_words$word,
         str_detect(word,"^[a-z']+$"))
```

Unnested table example (5 rows)

```
kable(df_reviews_words[1:5,]) %>%
  kable_styling("striped", full_width = F, latex_options = "HOLD_position")
```

asin	reviewID	overall	word
120401325X	1	4	stick
120401325X	1	4	rounded
120401325X	1	4	shape
120401325X	1	4	bumping
120401325X	1	4	siri

AFINN Analysis

The AFINN lexicon will be used to provide scores for words on a scale of -5 (most negative) to +5 (most positive)

Values for the AFINN lexicon

```
AFINN_lex_sent <- get_sentiments("afinn") %>%
  select(word, afinn_score = value)
```

Join AFINN to the tidy text table of Amazon reviews

```
AFINN_reviews_sentiment <- df_reviews_words %>%
  inner_join(AFINN_lex_sent, by = "word") %>%
  group_by(reviewID, overall) %>%
  summarize(sentiment = mean(afinn_score))
```

```
## `summarise()` has grouped output by 'reviewID'. You can override using the
## `.groups` argument.
```

Table preview

```
kable(AFINN_reviews_sentiment[1:10,]) %>%
  kable_styling("striped", full_width = F, latex_options = "HOLD_position")
```

reviewID	overall	sentiment
1	4	-3.000000
2	5	2.000000
3	5	4.000000
4	4	2.333333
5	5	3.000000
6	3	1.500000
7	5	2.000000
8	1	-1.000000
9	5	0.600000
10	5	0.800000

As we can see, now the results include a sentiment value along with their score

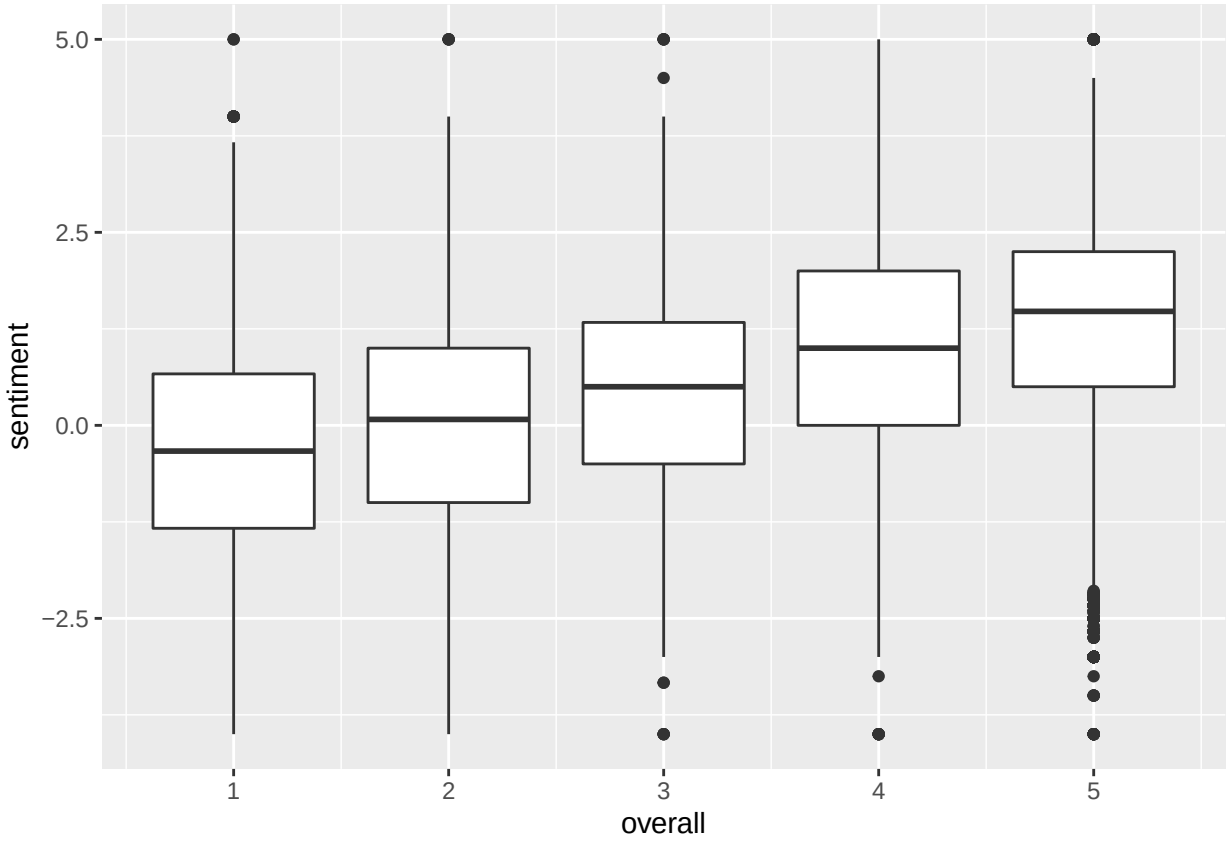
Summary table

```
dt_ars <- data.table(AFINN_reviews_sentiment)
dt_ars <- dt_ars[,list(mean=mean(sentiment),sd=sd(sentiment)),by=overall]
dt_ars[order(overall)]
```

```
##    overall        mean       sd
## 1:       1 -0.31523727 1.445546
## 2:       2  0.09505335 1.394675
## 3:       3  0.37678090 1.409068
## 4:       4  0.82910723 1.341593
## 5:       5  1.27168750 1.383566
```

Box plot

```
ggplot(AFINN_reviews_sentiment, aes(overall, sentiment, group = overall)) + geom_boxplot()
```



The plot shows clearly that there is a relative correlation between the aggregate AFINN sentiment score of a review and its star rating.

In the Yelp example this effect was more significant, maybe Amazon reviews are naturally more normalized?

Word Sentiment and Word Clouds

Most common words in all reviews

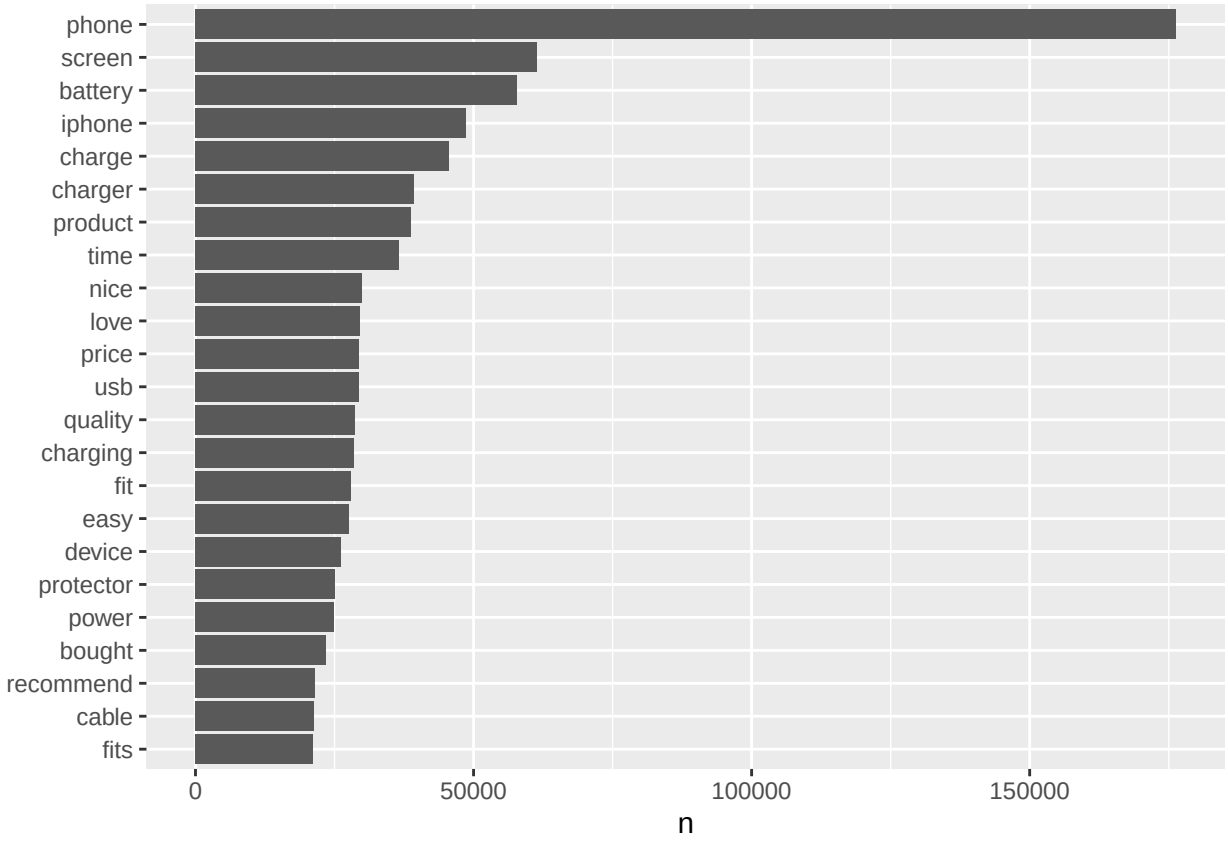
```
xtable(head(df_reviews_words %>%
              count(word, sort = TRUE))) %>%
  kable() %>%
  kable_styling(bootstrap_options = c("striped", "hover", "condensed"), full_width = F, latex_options =
```

word	n
phone	176371
screen	61439
battery	57856
iphone	48600
charge	45624
charger	39228

Plot common words with more than 20000 occurrences

```
df_reviews_words %>%
  count(word, sort = TRUE) %>%
  filter(n > 20000) %>%
```

```
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(word, n)) +
  geom_col() +
  xlab(NULL) +
  coord_flip()
```



Find most common positive and negative words

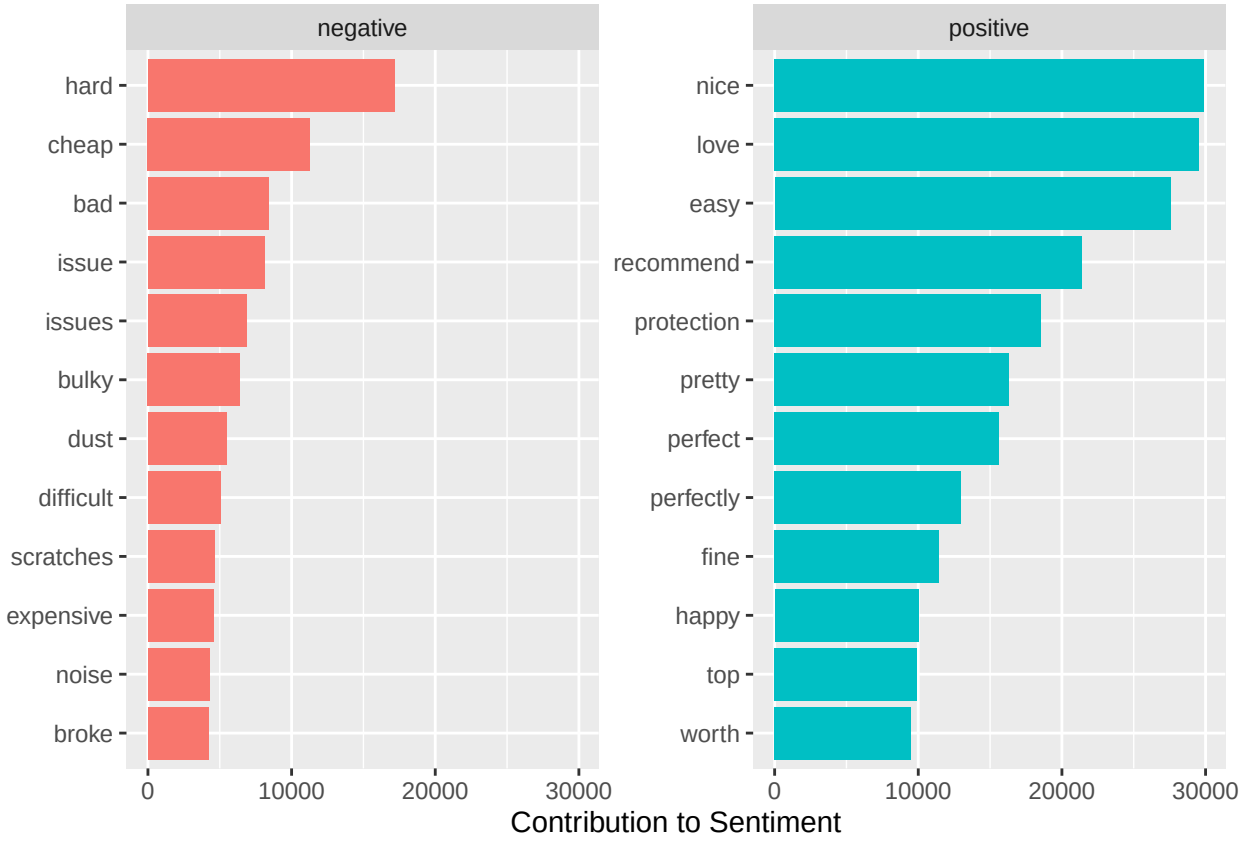
```
Sentiment_Analysis_Word_Count <- df_reviews_words %>%
  inner_join(get_sentiments("bing"), "word") %>%
  count(word, sentiment, sort = TRUE) %>%
  ungroup()
```

This plot clearly shows that **phone** is the most common word and the one with the highest weight which is correct as we saw before that this word had over 150.000 occurrences in the reviews; as much as three times more occurrences than **screen** which is the second one.

Plot the words and their contribution to the sentiment

```
Sentiment_Analysis_Word_Count %>%
  group_by(sentiment) %>%
  top_n(12, n) %>%
  ungroup() %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(word, n, fill = sentiment)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~sentiment, scales = "free_y") +
  labs(y = "Contribution to Sentiment", x = NULL) +
```

```
  coord_flip()
```

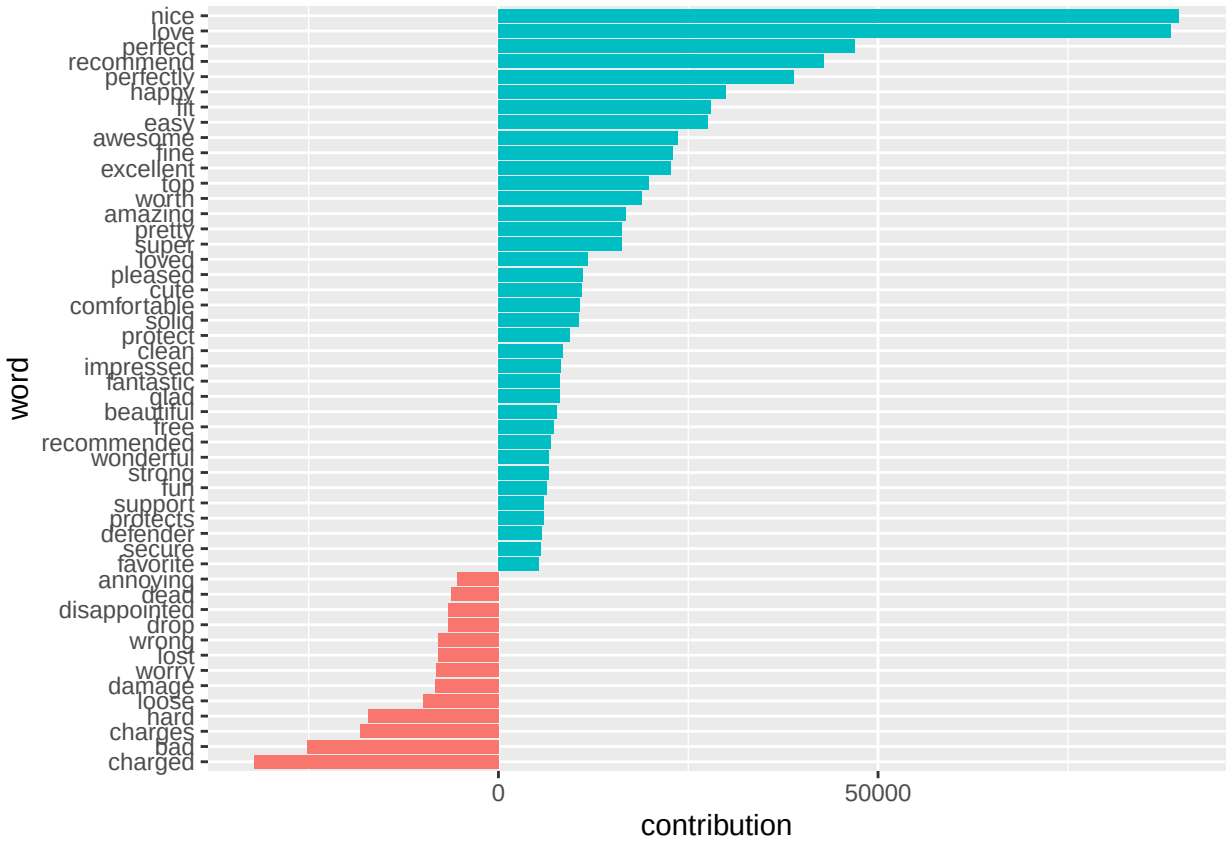


The most negative words in the reviews are **hard**, **cheap** and **bad** and the most positive are **nice**, **love** and **easy**. These results seem to indicate that the analysis is correct as these words are mostly what you would expect when choosing positive or negative words for a review.

Plot their relationship

```
Sentiment_Analysis_Word_Contribution <- df_reviews_words %>%
  inner_join(get_sentiments("afinn"), by = "word") %>%
  group_by(word) %>%
  summarize(occurences = n(), contribution = sum(value))

Sentiment_Analysis_Word_Contribution %>%
  top_n(50, abs(contribution)) %>%
  mutate(word = reorder(word, contribution)) %>%
  ggplot(aes(word, contribution, fill = contribution > 0)) +
  geom_col(show.legend = FALSE) +
  coord_flip()
```



Plot the word clouds

```
df_reviews_words %>%
  anti_join(stop_words, "word") %>%
  count(word) %>%
  with(wordcloud(word, n, max.words = 50))
```

Plot the word clouds grouped by sentiment

```
df_reviews_words %>%
  inner_join(get_sentiments("bing"), "word") %>%
  count(word, sentiment, sort = TRUE) %>%
  acast(word ~ sentiment, value.var = "n", fill = 0) %>%
  comparison.cloud(colors = c("gray20", "gray80"), max.words = 50)
```

```
## Warning in comparison.cloud(., colors = c("gray20", "gray80"), max.words = 50):
## protection could not be fit on page. It will not be plotted.
```

```
## Warning in comparison.cloud(., colors = c("gray20", "gray80"), max.words = 50):
## disappointed could not be fit on page. It will not be plotted.
```

Here **hard** and **nice/love** seem to be the dominant words for positive and negative sentiments.

Outlier Analysis

Given a dataset as big as this one, there will be cases where a review that has very few stars has a sentiment that is positive. This can be a mistake on the part of the user or done on purpose in order to be displayed at the top.

1. One star review with a sentiment > 3.70

```
AFINN_reviews_sentiment_outlier_1 <- AFINN_reviews_sentiment %>%
  filter(overall == 1 & sentiment > 3.70) %>%
  select(reviewID)

df_reviews_outlier_low <- df_reviews %>%
  select(reviewID,reviewText,overall) %>%
  filter(reviewID %in% as.list(AFINN_reviews_sentiment_outlier_1$reviewID))

kable(df_reviews_outlier_low) %>%
  kable_styling("striped", full_width = F, latex_options = "HOLD_position")
```

reviewID	reviewText
13812	Low volume wants to fall off. Typical problem with BT makes them unusable. I am a ups driver so the BT ne
44293	Worked amazing...for about 5 days! Then nothing! No juice, no charge absolutely nothing. Got what I paid fo
46264	If you value the wonderful screen that comes with the iPhone 4S, don't put this one on! It covers the screen w
57077	I was thrilled to get a white case until I found out that it is darn near impossible to get of your phone. I had t
83736	When i received this phone case, i was very excited. Mainly because a friend of mine has one of these and his a
98303	When the Battery came it was keeping the charge about 10 hrs. then I had to plug it in every 6hrs .!!! Then n
104737	ITS TOO SMALL WHO EVER MADE THESE DID NOT THINK ABOUT THE SIZE OF THE GALAXY!
110410	This case came right away and was made with quality but I prefer the pouch to carry my phone around becau
117063	Tried to install for Infinity Ipod connector ('08 G37) but no led's light up and no response to bluetooth from M
118729	I had the phone for 1 week before it told me i have not enough space left on the device.... Follow my advice if
138215	This is literally just a bumper and screen protectors. And the bumper was made of plastic... I was not very th
156027	It's funny how this product is awesome on iphone 5. Mine came with lint already under before placing on the
160401	This is funny case i ever seen ... it look like toy not cover .... so please dont buy....unless you like something lik
161905	This cable did not work with my iPhone 5s - phone said incompatible accessory. So what lightening device is t
165179	seemed awesome to get so many, but the thing is- they don't work. they don't charge for anything. okay for da
179815	Received the case early, but it already had scratches and fits funny. Not to mention, it's a small sticker inside a
184761	this is another product of cheap quality but hey what can you say you look at it on amazon and it looks good
189849	I received this charger and, while the indicator light on the charger illuminated, the charger did not charge my

There are many reviews that show an input error from the user and others in which the user dismisses the product bought but praises another product in the review.

2. Five star reviews with sentiment < 3.70

```
AFINN_reviews_sentiment_outlier_5 <- AFINN_reviews_sentiment %>%
  filter(overall == 5 & sentiment < -3.70) %>%
  select(reviewID)

df_reviews_outlier_high <- df_reviews %>%
  select(reviewID,reviewText,overall) %>%
  filter(reviewID %in% as.list(AFINN_reviews_sentiment_outlier_5$reviewID))

kable(df_reviews_outlier_high) %>%
  kable_styling("striped", full_width = F, latex_options = "HOLD_position")
```

reviewID	reviewText
10298	The the shit
21642	Definitely paid a whole hell of a lot less for used than new and it came in looking like brand new.
27935	This thing is a lifesaver.I currently use my iPhone 4s with a mophie juice pack air and no way in hell was it fit
40160	Had this on the wife's phone for two years, no problem charging and maintaining and held a long damn charge
53133	Tried two cheaper cables. One didn't work at all, and the other one was very finicky, couldn't move the tablet
55665	This thing works well and is tough as hell. Doesn't make the phone bulk up. Everyone now fears me and my p
65292	Were these speakers developed using alien technology? I can't believe how good they sound. How the hell did
79736	My wife's LG Motion (MetroPCS) fit's in it good and it holds it great.No looseness and keeps the phone in he
91986	It works and does what it's supposed to do. great replacement for the cheap ass factory charging coeds u get t
169437	There's not a lot to say about this kind of product except it works a lot better then the clamp on-suction cup
193518	The case is like a transformer to me. It is damn durable and live the design and price

The low score on the sentiment here is mostly due to profanities and expletives that the users write in order to praise the product but that are interpreted as having negative sentiment value

TF-IDF

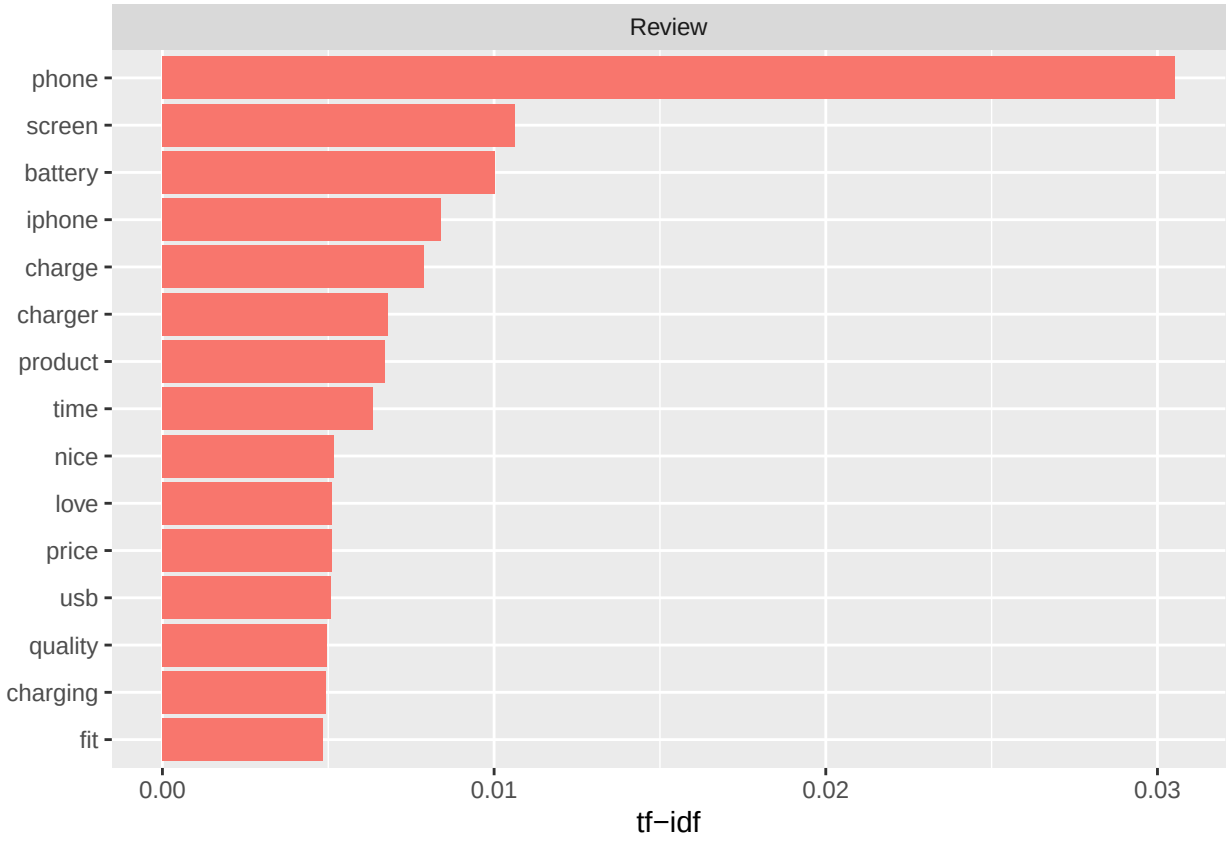
The idea of using TF(Term Frequency)-IDF(Inverse Document Frequency) is to find words that are important in the reviews but are not too common

Calculate TF-IDF

```
term_frequency_review <- df_reviews_words %>% count(word, sort = TRUE)
term_frequency_review$total_words <- as.numeric(term_frequency_review %>% summarize(total = sum(n)))
term_frequency_review$document <- as.character("Review")
term_frequency_review <- term_frequency_review %>%
bind_tf_idf(word, document, n)
```

Plot the results

```
term_frequency_review %>%
  arrange(desc(tf)) %>%
  mutate(word = factor(word, levels = rev(unique(word)))) %>%
  group_by(document) %>%
  top_n(15, tf) %>%
  ungroup() %>%
  ggplot(aes(word, tf, fill = document)) +
  geom_col(show.legend = FALSE) +
  labs(x = NULL, y = "tf-idf") +
  facet_wrap(~document, ncol = 2, scales = "free") +
  coord_flip()
```



As we can see, the three most important words that appear in the reviews are **phone**, **screen** and **battery**.

This result is almost the same as the values obtained in the chart that plotted the number of occurrences of words in the reviews. There might not be many words that validate the idea of the TF-IDF or the sheer number of occurrences of words such as **phone** dwarf their contribution.