Final Project

Team - A: Divya Bhattiprolu, Ekaterina Becker, Escamilla Karen

2020-08-27

Load required libraries

```
library(vroom)
library(tidyverse)
library(dplyr)
library(skimr)
library(janitor)
library(magrittr)
library(stringr)
library(tidytext)
library(wordcloud)
library(reshape2)
```

Team - A conducted text analysis of application reviews from customers in different countries. The data was derived from 27 excel documents containing 408,648 observations of 19 variables.

The combining of the 27 excel files were done by using the "vroom" package to create one data frame containing all the information for better analysis.

```
files <- fs::dir_ls(path = "C:/Users/srila/Downloads/FUNDAMENTALS OF BUSINESS ANLYT - 2019-2020 Summer
App_Data <- vroom::vroom(files)
```

```
## Rows: 408,648
## Columns: 17
## Delimiter: ","
## chr [15]: App Name, App Store, App, Store, App ID, Review ID, Country, Version, Date, Auth...
## dbl [ 1]: Rating
## lgl [ 1]: Device
##
## Use `spec()` to retrieve the guessed column specification
## Pass a specification to the `col_types` argument to quiet this message
```

Below is the number of reviews for each application ("app"). The app with the most reviews in our data set is "Maps - Navigation & Transit" with 207,057 reviews. The app with the least amount of reviews is "AcuraLink" with 9. To create insights for HERE apps we looked at 10,736 reviews between "HERE WeGo - Offline Maps and GPS" and "HERE WeGo - City Navigation".

```
count (App_Data, `App Name`, sort = TRUE)
```

```
## # A tibble: 7 x 2
```

```
##   `App Name`                                                n
##   <chr>                                                 <int>
## 1 Maps - Navigation & Transit                          207057
## 2 Waze - GPS, Maps, Traffic Alerts & Live Navigation 150547
## 3 Google Maps - GPS Navigation                          27594
## 4 Waze Navigation & Live Traffic                        12705
## 5 HERE WeGo - Offline Maps & GPS                        10008
## 6 HERE WeGo - City navigation                             728
## 7 AcuraLink                                                 9
```

In this section, we looked at descriptive statistics of all data that we have available for analysis.

```
skim(App_Data)
```

Table 1: Data summary

Name	App_Data
Number of rows	408648
Number of columns	17
Column type frequency:	
character	15
logical	1
numeric	1
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
App Name	0	1.00	9	50	0	7	0
App Store	0	1.00	3	11	0	2	0
App	0	1.00	9	50	0	7	0
Store	0	1.00	3	11	0	2	0
App ID	0	1.00	8	28	0	7	0
Review ID	0	1.00	10	120	0	403992	0
Country	0	1.00	3	21	0	139	0
Version	367621	0.10	3	8	0	69	0
Date	0	1.00	9	10	0	184	0
Author	100864	0.75	1	69	0	287741	0
Subject	362895	0.11	1	191	0	31129	0
Body	393	1.00	1	6000	0	271587	0
Translated Subject	386148	0.06	1	429	0	14467	0
Translated Body	302129	0.26	1	23109	0	78380	0
Emotion	349368	0.15	3	9	0	5	0

Variable type: logical

skim_variable	n_missing	complete_rate	mean	count
Device	408648	0	NaN	:

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
Rating	0	1	4.14	1.36	1	4	5	5	5	

Variable	Data Type
App Name	Nominal
App Store	Nominal
App ID	Continuous
Review ID	Continuous
Country	Nominal
Version	Continuous
Rating	Continuous
Date	Discrete
Author	Nominal
Subject	Nominal
Body	Nominal
Trans. Sub	Nominal
Trans. Body	Nominal
Emotion	Nominal
Device	

As we can see the majority of our variables are categorical, and only a few are quantitative. To show the descriptive statistics, we are choosing to look at the numeric variable: Rating.

Here is the snapshot of descriptive statistics for all apps combined:

```
mean(App_Data$Rating)
```

```
## [1] 4.137757
```

```
median(App_Data$Rating)
```

```
## [1] 5
```

```
sd(App_Data$Rating)
```

```
## [1] 1.357826
```

mean 4.14 median 5 sd 1.35

Here is a summary of the descriptive statistics for HERE apps:

```
##mean(App_Data_HERE_Only$Rating)
##median(App_Data_HERE_Only$Rating)
##sd(App_Data_HERE_Only$Rating)
```

mean 3.84 median 5 sd 1.5

We can see that HERE apps have a lower mean than the mean of all the other apps combined. In our research, we are going to determine what causes HERE apps have lower ratings than their competitors.

As mentioned before the app "AcuraLink" has the least amount of reviews compared to the other apps. As a team we collectively decided to remove this application from further analysis because its over all usefulness for our analysis is obsolete. Below is the code we used to remove the application from our data set.

```
App_Data_No_Acura <-App_Data[!(App_Data$`App Name`=="AcuraLink"),]
App_Data_No_Acura
```

```
## # A tibble: 408,639 x 17
##    `App Name` `App Store` App   Store `App ID` `Review ID` Country Version
##    <chr>      <chr>       <chr> <chr> <chr>    <chr>       <chr>   <chr>
##  1 Google Ma~ iOS         Goog~ iOS   5850273~ 2361429743  China   4.47
##  2 Google Ma~ iOS         Goog~ iOS   5850273~ 2361409552  China   4.47
##  3 Google Ma~ iOS         Goog~ iOS   5850273~ 2361279254  China   4.47
##  4 Google Ma~ iOS         Goog~ iOS   5850273~ 2361458244  USA     4.47
##  5 Google Ma~ iOS         Goog~ iOS   5850273~ 2361375637  USA     4.47
##  6 Google Ma~ iOS         Goog~ iOS   5850273~ 2361364277  USA     4.47
##  7 Google Ma~ iOS         Goog~ iOS   5850273~ 2361292138  Japan   4.47
##  8 Google Ma~ iOS         Goog~ iOS   5850273~ 2361495251  Austra~ 4.47
##  9 Google Ma~ iOS         Goog~ iOS   5850273~ 2361283307  Austra~ 4.47
## 10 Google Ma~ iOS         Goog~ iOS   5850273~ 2361421102  Vietnam 4.47
## # ... with 408,629 more rows, and 9 more variables: Rating <dbl>, Date <chr>,
## #   Author <chr>, Subject <chr>, Body <chr>, `Translated Subject` <chr>,
## #   `Translated Body` <chr>, Emotion <chr>, Device <lgl>
```

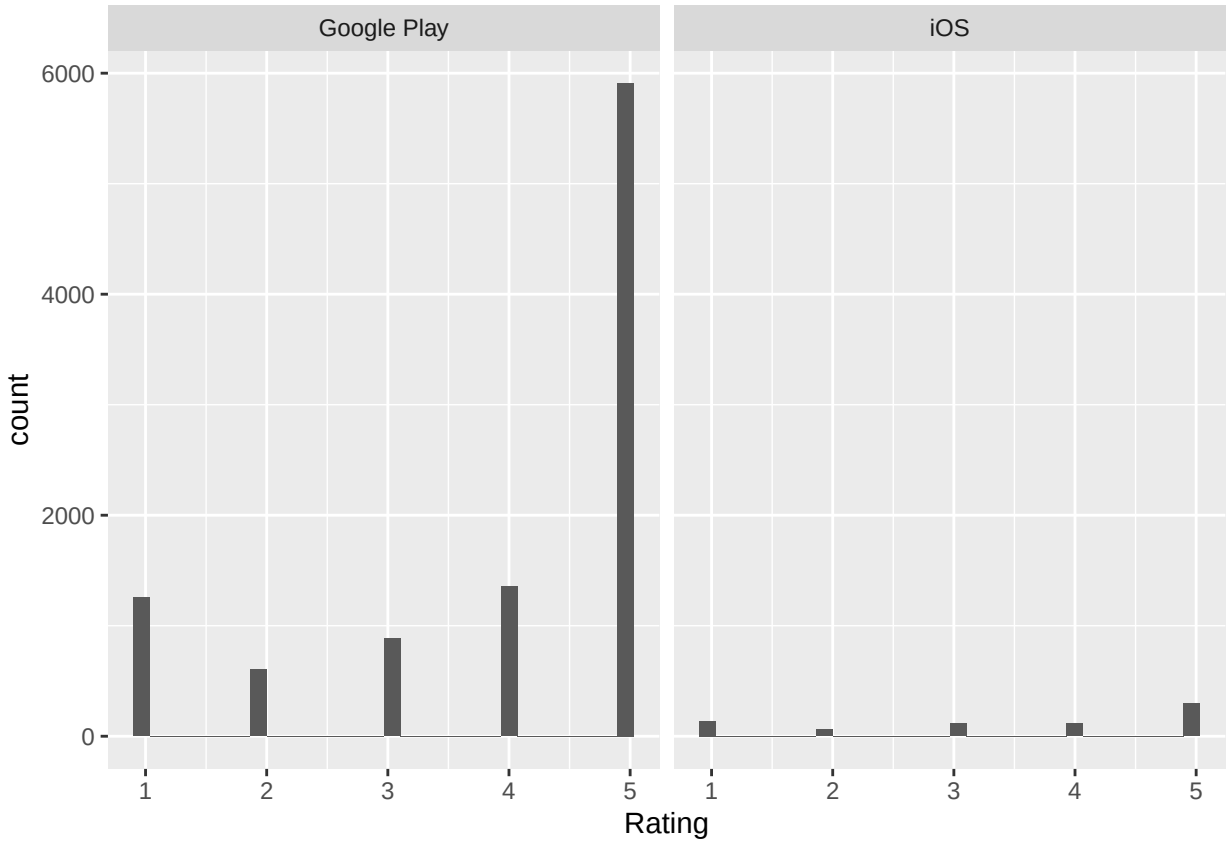
Because the data set contains empty cells or spaces we are using the package "Janitor" to clean up the data frame and remove them from the the overall information.

```
Clean_Spaces_App <- clean_names(App_Data_No_Acura)
```

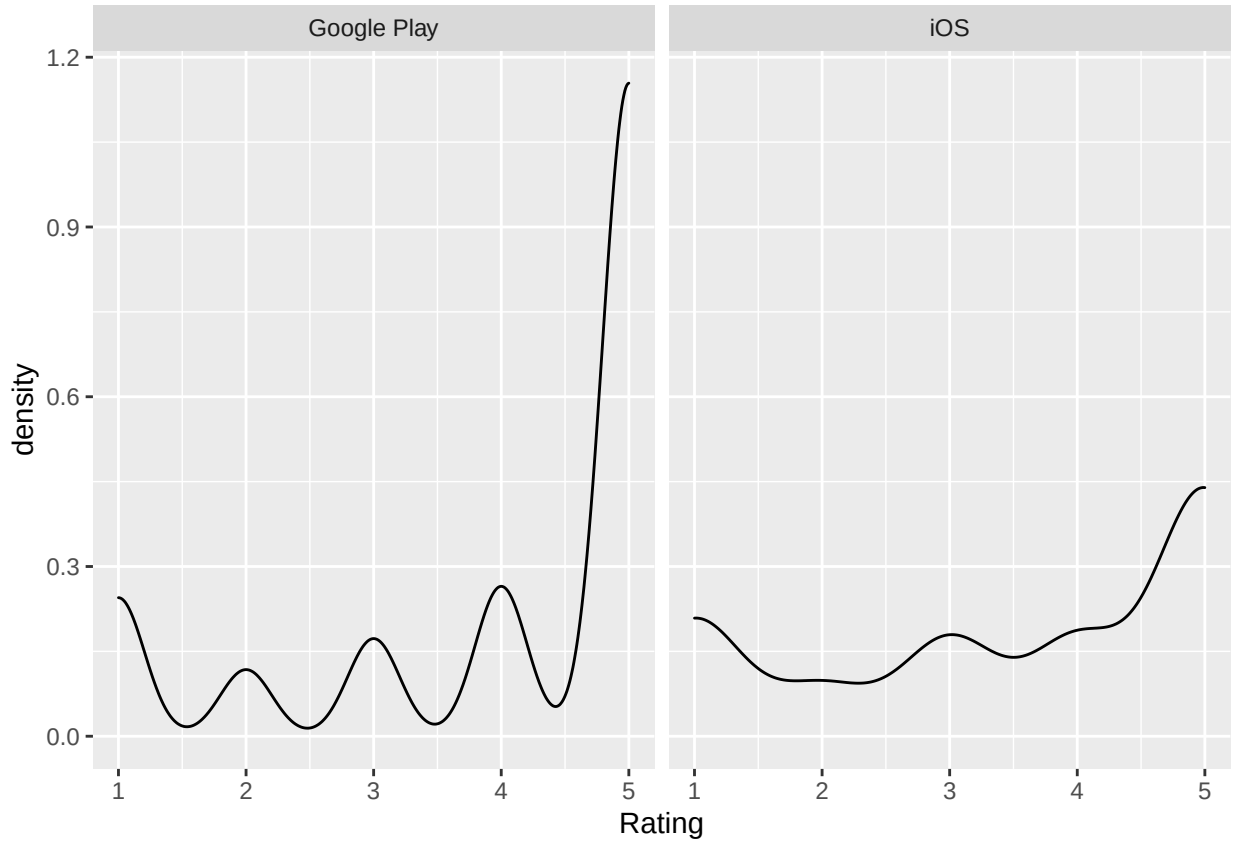
Below, we created a histogram showing the difference between the ratings for the apps "HERE" on Google and IOS.

```
App_Data_No_Acura %>% filter(str_detect(`App Name`,"HERE")) %>% ggplot(aes(x = Rating)) + geom_histogra
```

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




```
App_Data_No_Acura %>% filter(str_detect(`App Name`,"HERE")) %>% ggplot(aes(x = Rating)) + geom_density()
```



As we can see above, a similar trend can be found in both App Stores. While both store demonstrate a very strong 5 star rating, Google Play shows a much greater density than its competitor IOS.

An interesting trend can be seen for both app stores. Google Play shows an upward trend through out the ratings until it reaches to a rating of 5. For IOS the peaks are more fluid compared to Google Pay with a dramatic spike only between ratings 4 and 5.

In the next phase of reviewing the information provided by tokenizing our data to better understand the emotion or meaning of the text.

Because our data represents various countries, we have to account for reviews in different languages. We create an additional column that has both the Body information, which are the English reviews, and the Translated Body containing the reviews that were in other languages but were translated to English for better interpretation. We use the below the code to do this.

```
App_Data_No_Acura_Cleaned_Body <- App_Data_No_Acura %>% mutate(mycol = coalesce(`Translated Body`, Body)
```

Next, we want to know how many N/A reviews are in our data frame.

```
table(is.na(App_Data_No_Acura_Cleaned_Body$mycol))
```

```
##
##  FALSE   TRUE
## 408248    391
```

We can see that we only have 391 reviews in our new column that contain N/A. Since the column containing N/A represents only 0.095% of all reviews, we deem this not to affect our overall interpretation of the information.

Since we created a new column where all reviews are in the English language, we can take the next step for tokenizing our data.

```
App_Data_Unnest <- App_Data_No_Acura_Cleaned_Body %>%
  unnest_tokens(word, mycol)
```

By doing the above we have developed a new column in our data set called "word". This column breaks down every review by word and creates its own row. Because of this we have created 4,504,962 observations.

With a tokenized data set we can further look at the sentiment behind the text used in the reviews by applying lexicon packages.

The first lexicon we will be using to interpret the tokenized data set will be "afinn".

We can see below that the corresponding value of rating and afinn sentiment. The rating of 1 has a negative sentiment of -0.47, and a positive rating of 5 has a rating of 1.97, which proves that the tokenization is working, because negative values have a low rating and positive reviews have a high one.

```
afinn_df <-get_sentiments("afinn")

App_Data_Sentiment <- App_Data_Unnest %>%
  inner_join(afinn_df)
```

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

```
group_by(App_Data_Sentiment, Rating) %>% summarize(n=n(),total_value = sum(value)) %>% mutate(wt=total_
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 5 x 4
##   Rating      n total_value      wt
##    <dbl>  <int>       <dbl>   <dbl>
## 1      1  48511      -22769 -0.469
## 2      2  23704       -2341 -0.0988
## 3      3  34189       15220  0.445
## 4      4  53718       71653  1.33
## 5      5 220467      433696  1.97
```

We want to see how the above information translates into the different areas of our data set. Below we group the information by the application name, number of reviews and the percentage. Because we have inner joined our reviews with the afinn package we can see which app has a higher amount of positive reviews compared to the others.

```
group_by(App_Data_Sentiment, `App Name`) %>% summarize(n=n(),total_value = sum(value)) %>% mutate(wt=to
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

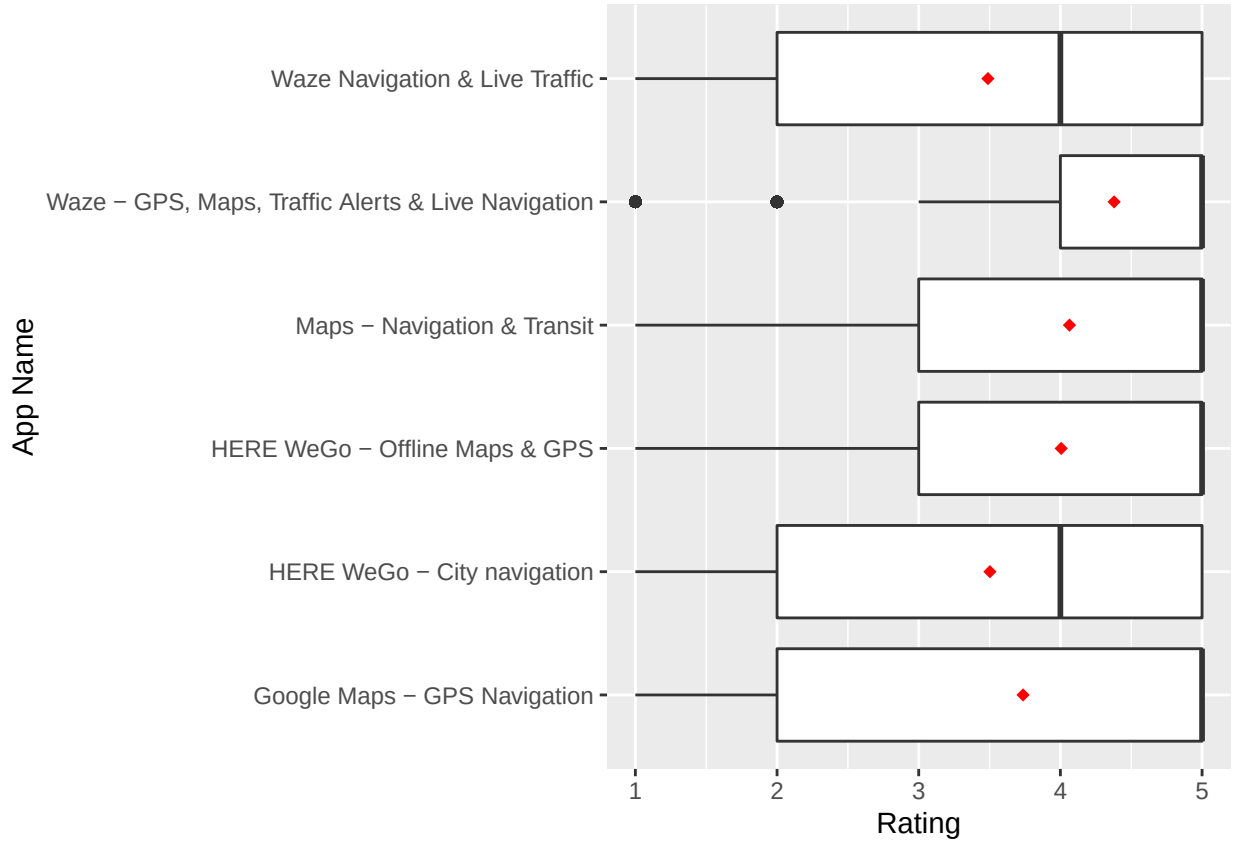
```
## # A tibble: 6 x 4
##   `App Name`                                      n total_value    wt
##   <chr>                                       <int>       <dbl> <dbl>
## 1 Google Maps - GPS Navigation                67320       53057 0.788
## 2 HERE WeGo - City navigation                  1600         861 0.538
```

```
## 3 HERE WeGo - Offline Maps & GPS                          9293        8745 0.941
## 4 Maps - Navigation & Transit                           159731      239035 1.50
## 5 Waze - GPS, Maps, Traffic Alerts & Live Navigation    109208      168701 1.54
## 6 Waze Navigation & Live Traffic                         33437       25060 0.749
```

By using the above code we can see "Waze - GPS, Maps, Traffic Alerts & Live Navigation" shows to have the most positive reviews, closely followed by Maps - Navigation & Transit. HERE WeGo - Offline Maps & GPS ranks 3rd among all apps.

This corresponds with the median ratings, which can be seen from the box plot.

```
App_Data_No_Acura %>% ggplot(aes(x = Rating, y = `App Name`, fill = Rating))+ geom_boxplot(position = "
```



Next we will create visuals for HERE apps using bing lexicon.

```
App_Data_Sentiment_bing <- App_Data_Unnest %>%
  inner_join(get_sentiments("bing"))
```

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

In the code below, we are creating a filter to only look at the top 10 words for HERE apps.

```
App_Data_HERE_Only <- App_Data_Sentiment_bing %>% filter(str_detect(`App Name`,"HERE"))
```

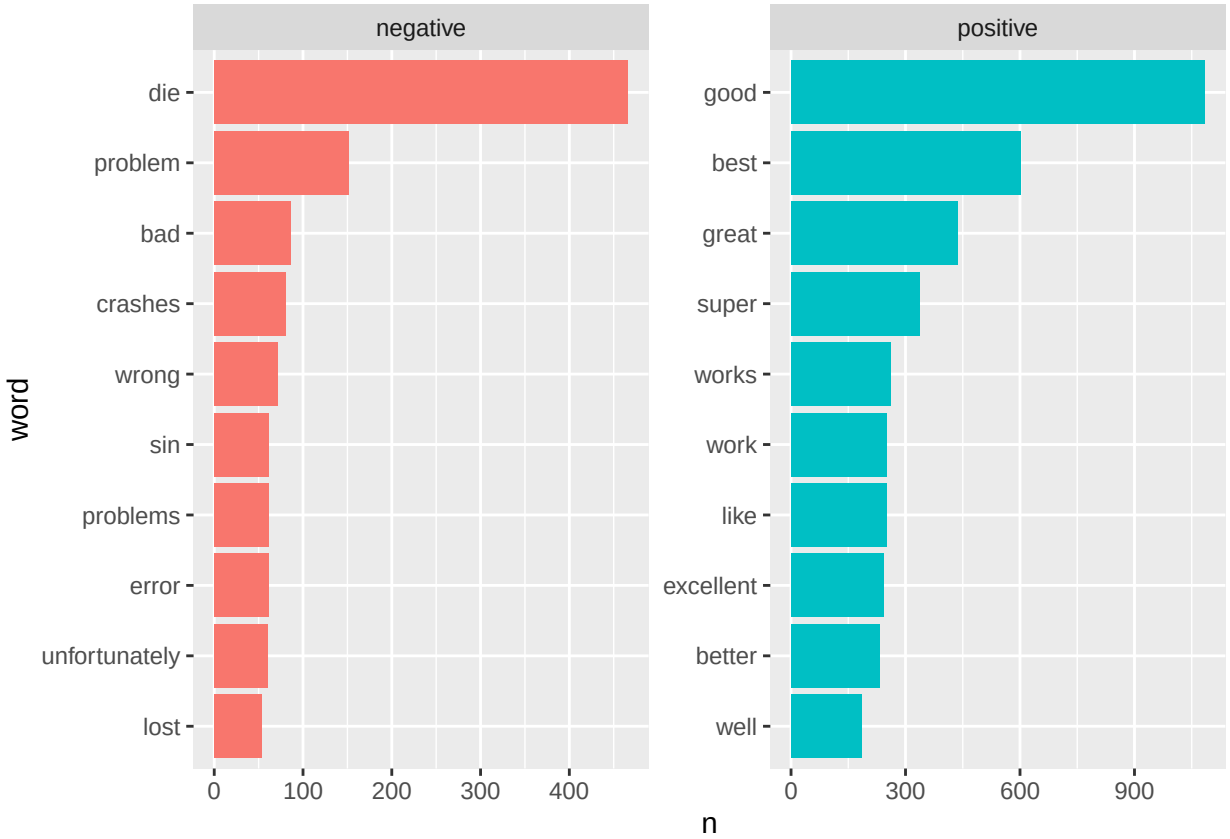
To create the visuals, we group by the sentiment and take the top 10 words for each sentiment. The words used are categorized in positive and negative connotations.

```
App_Data_Sentiment_Number <- App_Data_HERE_Only %>%
count(word, sentiment) %>%
group_by(sentiment) %>%
  top_n(10) %>%
  ungroup() %>%
  mutate(`word` = reorder(`word`, n))
```

```
## Selecting by n
```

Here, by using the ggplot we make a bar plot of the above information.

```
App_Data_Sentiment_Number %>% ggplot(aes(word, n, fill = sentiment)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~sentiment, scales = "free") +
  coord_flip()
```



The above bar plot shows the words that are the most commonly used in the categories of positive and negative.

We can see common themes among positive reviews, such as “good”, “works”, “well”. We can read that as the HERE applications work well for the users.

Among negative reviews, such as “die”, “problem”, “error” and “crashes”, we can suggest that the most common problem with HERE apps is that they crash often during usage.

We created a visual as a wordcloud to look at the most common words used to review both of HERE apps.

```
library(wordcloud)
App_Data_HERE_Only %>%
  anti_join(stop_words) %>%
  count(`word`) %>%
  with(wordcloud( word, n, max.words = 100))
```

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

The wordcloud demonstrates frequently used words in both the negative and positive sentiments. By reviewing the bar plot above and comparing the results of the wordcloud we can conclude those with the bigger font are those that are commonly used, as the font becomes smaller the word is used much less. We can see through this reasoning that many reviews had positive sentiments with words such as “super”, “excellent”, and “recommend”.

In addition to these words, we see can suggest the following:

Accurate - users enjoy the specificity and exactitude of the app Reliable - app demonstrates to be stable and dependable Faster - very agile software Convenient - app is considered to be useful and accessible Compatible - users find the app to be adaptable and in sync with other products Recommend - Word of mouth is very positive

Among negative words we can see “die” appears again as well as “wrong” and “bad”.

Further analysis leads us to concerning words such as,

Improvements - suggests that app contains glitches in the usage Crash - suggests that apps becomes unavailable for period of time Errors - applications have glitches while in usage

Next we are going to create a tokenized data set and filtering out each of the HERE apps. It's important to look at each of the apps separately to see if we can identify trends specific to each of the apps.

First, we tokenize each app separatelly:

```
App_Data_Unnest_HERE_Offline_Maps <- App_Data_Unnest %>% filter(str_detect(`App Name`,"HERE WeGo - Offli

App_Data_Unnest_City_Navigation <- App_Data_Unnest %>% filter(str_detect(`App Name`,"HERE WeGo - City na
```

Next, are looking at the lexicon of the top counties and assessing how HERE Offline Map is rated.

```
App_Data_Sentiment_HERE_Offline <- App_Data_Unnest_HERE_Offline_Maps %>%
  inner_join(afinn_df)
```

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

```
group_by(App_Data_Sentiment_HERE_Offline, Country)%>%summarize(n=n(),total_value = sum(value))%>% mutat
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 36 x 4
##    Country        n total_value     wt
##    <chr>      <int>       <dbl>  <dbl>
##  1 English     3563        4567  1.28
##  2 German      1269        -650 -0.512
##  3 Portuguese  1112        1177  1.06
##  4 Spanish      674         661  0.981
##  5 French       322         357  1.11
##  6 Polish       262         192  0.733
##  7 Russian      256         131  0.512
##  8 Italian      218         125  0.573
##  9 Serbian      168         314  1.87
## 10 Arabic       157         362  2.31
## # ... with 26 more rows
```

The top 4 languages that we have a largest number of reviews are English, German, Portuguese and Spanish. Among these languages English has a substantially higher number of reviews and it gets the top positive lexicon rating. The weight the English language carries in the afinn lexicon is 1.28.

In German, on the other hand, HERE Offline Map gets the weight of -0.51, meaning that there is a substantially higher number of negative reviews.

Considering a large number of reviews come from German speaking customer, we would recommend that HERE Offline Maps should conduct focus groups to study German consumers and gain a better understanding of the issues in that particular market.

We also see a strong positive trend in Arabic, however the number of reviews are limited. We recommend HERE Offline Maps to conduct a focus group to further investigate and develop better knowledge of the Arabic market.

Here is the visual of the top lexicons for HERE Offline Maps

```
App_Data_Sentiment_HERE_Offline_bing <- App_Data_Unnest_HERE_Offline_Maps %>%
  inner_join(get_sentiments("bing"))
```

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

```
App_Data_Sentiment_HERE_Offline_bing %>%
  count(word,sentiment, sort = TRUE) %>%
  acast( word ~ sentiment, value.var = "n", fill = 0) %>%
  comparison.cloud(colors = c("blue4", "red2"), max.words = 100)
```

These word trends appear to be consistent with the word trends for both HERE apps combined, such as crashing, problem and mistake for the negative reviews and accurate, reliable and fast for positive.

Below are the lexicons of the top countries and assessment of how HERE City Navigation is rated.

```
App_Data_Sentiment_HERE_City <- App_Data_Unnest_City_Navigation %>%
  inner_join(afinn_df)
```

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

```
group_by(App_Data_Sentiment_HERE_City, Country)%>%summarize(n=n(),total_value = sum(value))%>% mutate(w
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 55 x 4
##    Country            n total_value     wt
##    <chr>          <int>       <dbl>  <dbl>
##  1 Germany          445        -155 -0.348
##  2 USA              272         198  0.728
##  3 United Kingdom    80          85  1.06
##  4 France            68          40  0.588
##  5 Australia         58          50  0.862
##  6 Italy             57          71  1.25
##  7 Netherlands       53          62  1.17
##  8 Turkey            52          15  0.288
##  9 Canada            45          36  0.8
## 10 India             37          65  1.76
## # ... with 45 more rows
```

We see a similar trend for HERE City Navigation in Germany as we have seen for HERE Offline Maps, which demonstrates a negative sentiment.

We recommend a similar approach to HERE City Navigation as we did to HERE Offline Maps, conducting focus group studies for German consumers to gain a better understanding of the issues in that market.

Specifically for HERE City Navigation, we would recommend additional research to understand the difference in app performance between UK and USA consumers. We see that in the UK the performance appears more positive than in the USA. This can be a result of the great difference in amount of reviews between the two countries. The UK contained a total of 80 reviews, where the USA totaled 272. However we highlight these differences because both markets use a similar language, we believe further research might give us a better understanding in this particular market.

Below are the top lexicons for HERE City Navigation.

```
App_Data_Sentiment_HERE_City_bing <- App_Data_Sentiment_HERE_City %>%
  inner_join(get_sentiments("bing"))
```

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

```
App_Data_Sentiment_HERE_City_bing %>%
  count(word, sentiment, sort = TRUE) %>%
  acast(word ~ sentiment, value.var = "n", fill = 0) %>%
  comparison.cloud(colors = c("blue4", "red2"), max.words = 100)
```

negative

positive

We can see very similar trends for both applications. Positive words that appear in both apps are words such as recommend and stable. Words with a negative sentiment that reoccurs in both applications are those such as die, and crash. The word that we haven't see in HERE Offline Maps but see with HERE City Navigation is confusing. This suggests that consumers found this app more confusing than HERE Offline Maps.

Based on this, we recommend conducting further research to better understand what users find confusing about the app and potentially invest in end-user experience.

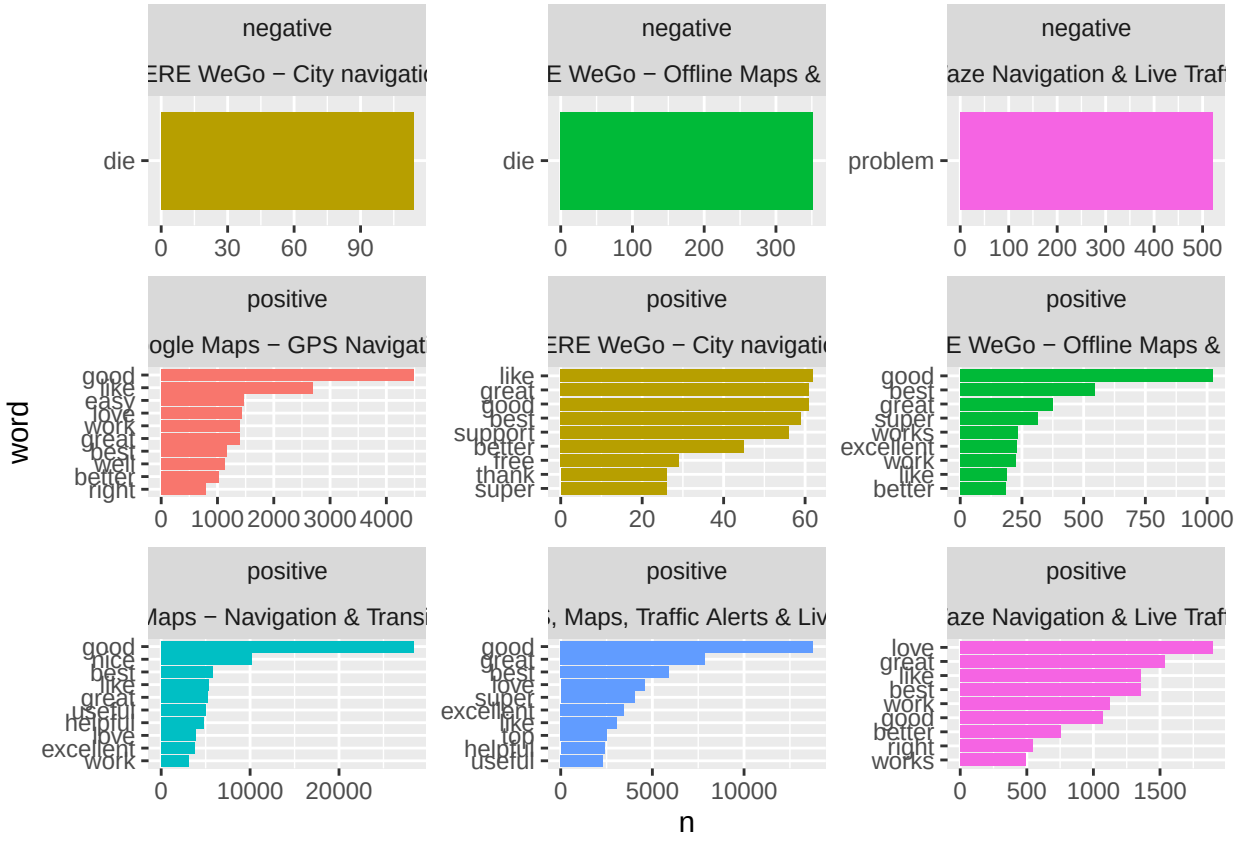
Next, we want to look at the lexicon reviews of all apps in our data set to see if anything different would pop out.

```
App_Data_Sentiment_AppNames <- App_Data_Sentiment_bing %>%
  count(word, sentiment, `App Name`) %>%
  group_by(`App Name`) %>%
  top_n(10) %>%
  ungroup() %>%
  mutate(word = reorder(paste(word, `App Name`, sep = "__"), n))
```

```
## Selecting by n
```

Now using ggplot, we plot the top 10 words of the positive and negative sentiments of all apps. We can see for both apps positive reoccurring words with a high frequency are words such as "good", "best" and "better". Whats most impressive is the one negative word that appears in both apps which is "die". The frequency of over 400 instances combined between both apps is a concern that may need further exploration.

```
App_Data_Sentiment_AppNames %>% ggplot(aes(x= word, y = n, fill = `App Name`)) +
 geom_col(show.legend = FALSE) +
 scale_x_discrete(labels = function(x) gsub("__.+$", "", x)) +
 facet_wrap(~ sentiment + `App Name` , nrow = 3, scale = "free") +
 coord_flip()
```



For the most part, in the positive lexicon we see words like useful and helpful. That suggests that the main reason why consumers use apps is to help them find navigation. The negative ratings for all apps, are the word "die" which suggests that consumers would rate the app negatively if it tends to have problems and if it dies.

Now we know what consumers are looking for and some of the issues with HERE apps. We are choosing to look at the German and English speaking consumers to derive insights for HERE.

```
App_Data_German <- App_Data_Sentiment_bing %>% filter(str_detect(`Country`,"German"))
```

```
App_Data_English <- App_Data_Sentiment_bing %>% filter(str_detect(`Country`,"English"))
```

Looking at visuals for "HERE" in German

```
App_Data_German %>%
  count(word, sentiment, sort = TRUE) %>%
  acast(word ~ sentiment, value.var = "n", fill = 0) %>%
  comparison.cloud(colors = c("blue4", "red2"),
                   max.words = 100)
```

Based on the visual for HERE apps in Germany, we see the following themes:

Crashes, problem, limit, freezes, unreliable, confusing, slow, bugs

We can conclude the app tends to crash and is slow at times. We would recommend that HERE invests in engineering research to account for these problems.

Looking at visuals for "HERE" in English

```
App_Data_English %>%
  count(word, sentiment, sort = TRUE) %>%
  acast(word ~ sentiment, value.var = "n", fill = 0) %>%
  comparison.cloud(colors = c("blue4", "red2"),
                   max.words = 100)
```

In English, unlike the visual shape in German, we see that the word "good" takes the center, while in German, the word "die" is featured. The positive themes in English speaking reviews are:

Accurate, useful, fast, reliable

Based on these themes we see English speaking consumers find the app enjoyable because of its dependability and easy usage. We would also recommend conducting a Net Promoter Score in the English speaking market to ensure that it ranks high to validate the positive analysis.

Considering that app is already performing well in these markets, we recommend that after validating the NPS, HERE also invests in advertising for English speaking consumers to promote and reach new customers.

We saw that Waze - GPS, Maps, Traffic Alerts & Live Navigation has the most positive reviews. We are analyzing this app to provide insights for HERE apps.

```
App_Data_Maps_Waze <- App_Data_Sentiment_bing %>% filter(str_detect(`App Name`,"Waze - GPS, Maps, Traffi
```

We created worldcloud and rshape for Waze - GPS, Maps, Traffic Alerts & Live Navigation to gain competitive insights

```
library(wordcloud)
App_Data_Maps_Waze %>%
  anti_join(stop_words) %>%
  count(`word`) %>%
  with(wordcloud(word, n, max.words = 100))
```

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


```
library(reshape2)
App_Data_Maps_Waze %>%
  count(word, sentiment, sort = TRUE) %>%
  acast(word ~ sentiment, value.var = "n", fill = 0) %>%
  comparison.cloud(colors = c("blue4", "violetred4"),
                   max.words = 100)
```

When looking at the app with the most number of positive reviews of Waze, we see very similar trends to positive reviews in HERE, such as the word “accurate”. This suggests that one of the most important aspects of map for navigation is accuracy. In addition, we see a new word pop up - intuitive. This leads us to think that in addition to accuracy and speed, the app’s intuitiveness is of value to the consumer. We suggest investing in user experience design for HERE apps to gain a better understanding of what consumers mean by intuitiveness and incorporate it to the overall design of “HERE”.

Among negatives, we see again a very similar trend to HERE apps with words such as “crash” and “errors”.

We can conclude that having a fast, reliable, and user friendly application that doesn’t crash is what consumers are looking for.