Sentiment Analysis - Airbnb Reviews

Suyash Tiwari

Table of Contents

#install.packages("qdap")  
library("qdap")

## Warning: package 'qdap' was built under R version 3.4.3

## Loading required package: qdapDictionaries

## Loading required package: qdapRegex

## Warning: package 'qdapRegex' was built under R version 3.4.3

## Loading required package: qdapTools

## Warning: package 'qdapTools' was built under R version 3.4.3

## Loading required package: RColorBrewer

##   
## Attaching package: 'qdap'

## The following object is masked from 'package:base':  
##   
## Filter

#install.packages("ggplot2")  
library("ggplot2")

## Warning: package 'ggplot2' was built under R version 3.4.3

##   
## Attaching package: 'ggplot2'

## The following object is masked from 'package:qdapRegex':  
##   
## %+%

#install.packages("ggthemes")  
library("ggthemes")

## Warning: package 'ggthemes' was built under R version 3.4.3

#install.packages("tidytext")  
library(tidytext)

## Warning: package 'tidytext' was built under R version 3.4.3

#install.packages("tidyr")  
library("tidyr")

## Warning: package 'tidyr' was built under R version 3.4.3

##   
## Attaching package: 'tidyr'

## The following object is masked from 'package:qdap':  
##   
## %>%

#install.packages("dplyr")  
library(dplyr)

## Warning: package 'dplyr' was built under R version 3.4.3

##   
## Attaching package: 'dplyr'  
##   
## The following object is masked from 'package:qdap':  
##   
## %>%

## The following object is masked from 'package:qdapTools':  
##   
## id

## The following object is masked from 'package:qdapRegex':  
##   
## explain

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

#install.packages("tm")  
library("tm")

## Warning: package 'tm' was built under R version 3.4.3

## Loading required package: NLP

##   
## Attaching package: 'NLP'

## The following object is masked from 'package:ggplot2':  
##   
## annotate

## The following object is masked from 'package:qdap':  
##   
## ngrams

##   
## Attaching package: 'tm'

## The following objects are masked from 'package:qdap':  
##   
## as.DocumentTermMatrix, as.TermDocumentMatrix

#install.packages("wordcloud")  
library("wordcloud")

## Warning: package 'wordcloud' was built under R version 3.4.3

bos\_reviews<-readRDS("C:/Users/suyas/Downloads/TwitteR/Datacamp/Sentiment Analysis/Sentiment Analysis - Airbnb Reviews/Sentiment-Analysis---Airbnb-Reviews/bos\_reviews.rds")  
head(bos\_reviews)

## id  
## 1 1  
## 2 2  
## 3 3  
## 4 4  
## 5 5  
## 6 6  
## comments  
## 1 My daughter and I had a wonderful stay with Maura. She kept in close touch with us throughout the day as we weren't arriving til later in the evening. The room was charming and the whole apartment was very warm and eclectic. She asked us what time we'd like breakfast and laid out a very nice spread. Right down the street from Jamaica Pond and very close to Center St. as well. All in all, great experience!  
## 2 We stay at Elizabeth's place for 3 nights in October 2014.\nThe apartment is really a great place to stay. \nLovely decorated and extremely well located. Very close to Back Bay station if you come by train, and close also to the subway and just a nice walk from the city center. \n\nI would say that is better that what you appreciate in the pictures\nAlso the bed is very confortable.\nSo really it was an optimal choice for us.  
## 3 If you're staying in South Boston, this is a terrific place to camp out. The apartment and bedroom are lovely, Ellie is an excellent host, and there is a lot within walking distance in a neighborhood on the rise.  
## 4 Derian and Brian were great and prompt with their communications with us. The room was as described; it was a small nice and clean room with a very comfortable bed and pillows. We shared a bathroom with others in the apartment. Derian had turned on the AC in our room prior to our arrival which we appreciated a lot. He also greeted us kindly when we arrived and showed us around. We arrived late and left early in the morning to catch our flight, so unfortunately, we did not get a chance to meet Brian or to talk much with Derian.   
## 5 John and Dan were gracious hosts and the location and accommodations were very nice......as listed.  
## 6 The best thing about Sean's place is the location. It's by a T station, there are a number of groceries and restaurants around, etc. It was nice to see all the drawings he made while studying architecture. The building is a bit old so walls are a bit thinner but overall it was a nice place to be.\n\nSean was great in answering phone calls and questions. He was there to meet us when we got to the place.

str(bos\_reviews)

## 'data.frame': 1000 obs. of 2 variables:  
## $ id : int 1 2 3 4 5 6 7 8 9 10 ...  
## $ comments: chr "My daughter and I had a wonderful stay with Maura. She kept in close touch with us throughout the day as we wer"| \_\_truncated\_\_ "We stay at Elizabeth's place for 3 nights in October 2014.\nThe apartment is really a great place to stay. \nLo"| \_\_truncated\_\_ "If you're staying in South Boston, this is a terrific place to camp out. The apartment and bedroom are lovely, "| \_\_truncated\_\_ "Derian and Brian were great and prompt with their communications with us. The room was as described; it was a s"| \_\_truncated\_\_ ...

dim(bos\_reviews)

## [1] 1000 2

# Practice apply polarity to first 6 reviews  
practice\_pol <- polarity(bos\_reviews$comments[1:6])

## Warning in polarity(bos\_reviews$comments[1:6]):   
## Some rows contain double punctuation. Suggested use of `sentSplit` function.

# Review the object  
practice\_pol

## all total.sentences total.words ave.polarity sd.polarity stan.mean.polarity  
## 1 all 6 390 0.747 0.398 1.875

# Check out the practice polarity  
summary(practice\_pol$all$polarity)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.2500 0.5009 0.6594 0.7466 1.0779 1.2455

# polarity for all reviews 1000  
bos\_pol <- polarity(bos\_reviews$comments)

## Warning in polarity(bos\_reviews$comments):   
## Some rows contain double punctuation. Suggested use of `sentSplit` function.

bos\_pol

## all total.sentences total.words ave.polarity sd.polarity stan.mean.polarity  
## 1 all 1000 70481 0.902 0.502 1.799

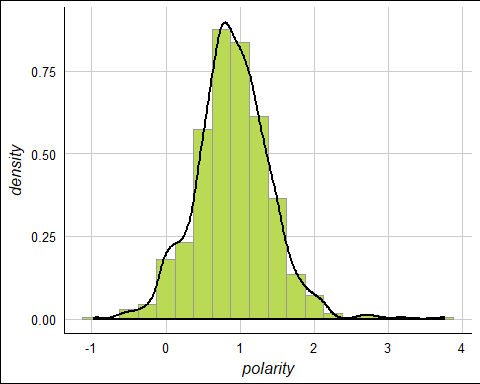
# Summary for all reviews  
summary(bos\_pol$all$polarity)

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## -0.9712 0.6047 0.8921 0.9022 1.2063 3.7510 1

# Plot it  
ggplot(bos\_pol$all, aes(x = polarity, y = ..density..)) +  
 theme\_gdocs() +   
 geom\_histogram(binwidth = 0.25, fill = "#bada55", colour = "grey60") +  
 geom\_density(size = 0.75)

## Warning: Removed 1 rows containing non-finite values (stat\_bin).

## Warning: Removed 1 rows containing non-finite values (stat\_density).



# Review  
bos\_pol$group

## all total.sentences total.words ave.polarity sd.polarity  
## 1 all 1000 70481 0.9021735 0.5015318  
## stan.mean.polarity  
## 1 1.798836

# Add polarity column  
bos\_reviews\_with\_pol <- bos\_reviews %>%   
 mutate(polarity = bos\_pol$all$polarity)  
  
# Subset positive comments   
pos\_comments <- bos\_reviews\_with\_pol %>%   
 filter(polarity > 0) %>%   
 pull(comments)  
  
# Subset negative comments  
neg\_comments <- bos\_reviews\_with\_pol %>%   
 filter(polarity<0) %>%   
 pull(comments)  
  
# Paste and collapse the positive comments  
pos\_terms <- paste(pos\_comments, collapse = " ")  
  
# Paste and collapse the negative comments  
neg\_terms <- paste(neg\_comments,collapse = " ")  
  
# Concatenate the terms  
all\_terms <- c(pos\_terms, neg\_terms)  
  
# Pipe a VectorSource Corpus  
all\_corpus <- all\_terms %>%   
 VectorSource() %>%   
 VCorpus()  
  
# Simple TFIDF TDM  
all\_tdm <- TermDocumentMatrix(  
 all\_corpus,   
 control = list(  
 weighting = weightTfIdf,   
 removePunctuation = TRUE,   
 stopwords = stopwords(kind = "en")  
 )  
)  
  
# Examine the TDM  
all\_tdm

## <<TermDocumentMatrix (terms: 4967, documents: 2)>>  
## Non-/sparse entries: 4350/5584  
## Sparsity : 56%  
## Maximal term length: 93  
## Weighting : term frequency - inverse document frequency (normalized) (tf-idf)

# Vector to tibble  
tidy\_reviews <- bos\_reviews %>%   
 unnest\_tokens(word, comments)  
  
# Group by and mutate  
tidy\_reviews <- tidy\_reviews %>%   
 group\_by(id) %>%   
 mutate(original\_word\_order = seq\_along(word))  
  
# Quick review  
tidy\_reviews

## # A tibble: 70,986 x 3  
## # Groups: id [1,000]  
## id word original\_word\_order  
## <int> <chr> <int>  
## 1 1 my 1  
## 2 1 daughter 2  
## 3 1 and 3  
## 4 1 i 4  
## 5 1 had 5  
## 6 1 a 6  
## 7 1 wonderful 7  
## 8 1 stay 8  
## 9 1 with 9  
## 10 1 maura 10  
## # ... with 70,976 more rows

# Load stopwords  
data("stop\_words")  
  
# Perform anti-join  
tidy\_reviews\_without\_stopwords <- tidy\_reviews %>%   
 anti\_join(stop\_words)

## Joining, by = "word"

# Get the correct lexicon  
bing<-get\_sentiments("bing")  
  
# Calculate polarity for each review  
pos\_neg <- tidy\_reviews %>%   
 inner\_join(bing) %>%  
 count(sentiment) %>%  
 spread(sentiment, n, fill = 0) %>%   
 mutate(polarity = positive - negative)

## Joining, by = "word"

# Check outcome  
summary(pos\_neg)

## id negative positive polarity   
## Min. : 1 Min. : 0.0000 Min. : 0.000 Min. :-10.000   
## 1st Qu.: 251 1st Qu.: 0.0000 1st Qu.: 4.000 1st Qu.: 3.000   
## Median : 499 Median : 0.0000 Median : 6.000 Median : 5.000   
## Mean : 500 Mean : 0.6633 Mean : 6.569 Mean : 5.906   
## 3rd Qu.: 748 3rd Qu.: 1.0000 3rd Qu.: 8.000 3rd Qu.: 8.000   
## Max. :1000 Max. :14.0000 Max. :42.000 Max. : 37.000

# Review tidy\_reviews  
tidy\_reviews

## # A tibble: 70,986 x 3  
## # Groups: id [1,000]  
## id word original\_word\_order  
## <int> <chr> <int>  
## 1 1 my 1  
## 2 1 daughter 2  
## 3 1 and 3  
## 4 1 i 4  
## 5 1 had 5  
## 6 1 a 6  
## 7 1 wonderful 7  
## 8 1 stay 8  
## 9 1 with 9  
## 10 1 maura 10  
## # ... with 70,976 more rows

# Review pos\_neg  
pos\_neg

## # A tibble: 977 x 4  
## # Groups: id [977]  
## id negative positive polarity  
## <int> <dbl> <dbl> <dbl>  
## 1 1 0 8 8  
## 2 2 0 7 7  
## 3 3 0 3 3  
## 4 4 1 7 6  
## 5 5 0 2 2  
## 6 6 0 5 5  
## 7 7 0 7 7  
## 8 8 0 3 3  
## 9 9 0 5 5  
## 10 10 1 18 17  
## # ... with 967 more rows

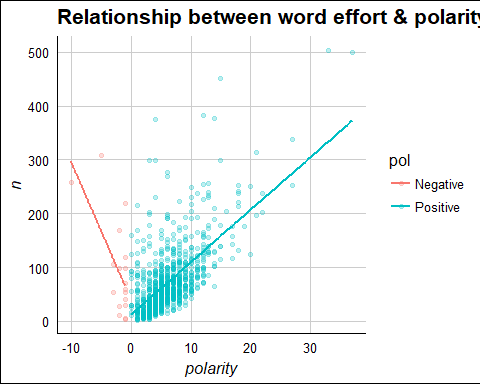
# Create effort  
effort<- tidy\_reviews %>%  
count(id)  
  
# Inner join  
pos\_neg\_with\_effort <- pos\_neg %>%  
inner\_join(effort)

## Joining, by = "id"

# Review   
pos\_neg\_with\_effort

## # A tibble: 977 x 5  
## # Groups: id [?]  
## id negative positive polarity n  
## <int> <dbl> <dbl> <dbl> <int>  
## 1 1 0 8 8 77  
## 2 2 0 7 7 80  
## 3 3 0 3 3 39  
## 4 4 1 7 6 101  
## 5 5 0 2 2 16  
## 6 6 0 5 5 79  
## 7 7 0 7 7 45  
## 8 8 0 3 3 27  
## 9 9 0 5 5 26  
## 10 10 1 18 17 117  
## # ... with 967 more rows

# Add pol  
pos\_neg\_pol <- pos\_neg\_with\_effort %>%  
 mutate(  
 pol = ifelse(  
 polarity >= 0,   
 "Positive",   
 "Negative"  
 )  
 )  
  
# Plot  
ggplot(  
 pos\_neg\_pol,   
 aes(polarity, n, color = pol)  
) +   
 geom\_point(alpha = 0.25) +  
 geom\_smooth(method = "lm", se = FALSE) +  
 theme\_gdocs() +  
 ggtitle("Relationship between word effort & polarity")



# Matrix  
all\_tdm\_m <- as.matrix(all\_tdm)  
  
# Column names  
colnames(all\_tdm\_m) <- c("positive","negative")  
  
# Top pos words  
order\_by\_pos <- order(all\_tdm\_m[, 1], decreasing = TRUE)  
  
# Review top 10 pos words  
all\_tdm\_m[order\_by\_pos, ] %>% head(n=10)

## Docs  
## Terms positive negative  
## walk 0.004557696 0  
## definitely 0.004172956 0  
## staying 0.003729024 0  
## city 0.003285093 0  
## wonderful 0.003107520 0  
## restaurants 0.003048329 0  
## highly 0.002959543 0  
## station 0.002693184 0  
## enjoyed 0.002426825 0  
## subway 0.002397230 0

# Top neg words  
order\_by\_neg <- order(all\_tdm\_m[,2], decreasing = TRUE)  
  
# Review top 10 neg words  
all\_tdm\_m[order\_by\_neg, ] %>% head(n=10)

## Docs  
## Terms positive negative  
## condition 0 0.002159827  
## don´t 0 0.002159827  
## demand 0 0.001439885  
## disappointed 0 0.001439885  
## dumpsters 0 0.001439885  
## hygiene 0 0.001439885  
## inform 0 0.001439885  
## it´s 0 0.001439885  
## nasty 0 0.001439885  
## safety 0 0.001439885

# Comparison cloud  
comparison.cloud(  
 all\_tdm\_m,   
 max.words = 20,  
 colors = c("darkgreen","darkred")  
)

## Warning in comparison.cloud(all\_tdm\_m, max.words = 20, colors =  
## c("darkgreen", : dumpsters could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 20, colors =  
## c("darkgreen", : hygiene could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 20, colors =  
## c("darkgreen", : inform could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 20, colors =  
## c("darkgreen", : nasty could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 20, colors =  
## c("darkgreen", : safety could not be fit on page. It will not be plotted.

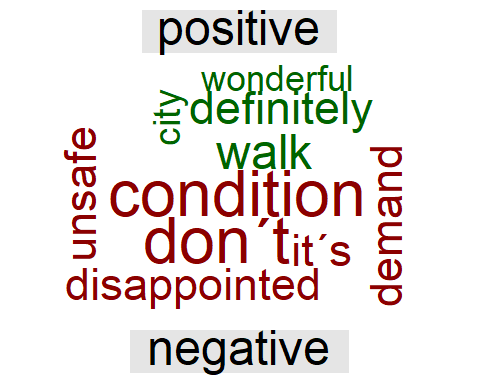
## Warning in comparison.cloud(all\_tdm\_m, max.words = 20, colors =  
## c("darkgreen", : shouldve could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 20, colors =  
## c("darkgreen", : sounds could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 20, colors =  
## c("darkgreen", : speaking could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 20, colors =  
## c("darkgreen", : staying could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 20, colors =  
## c("darkgreen", : restaurants could not be fit on page. It will not be  
## plotted.



# Review  
bos\_pol$all[1:6,1:3]

## all wc polarity  
## 1 all 77 1.1851900  
## 2 all 78 1.2455047  
## 3 all 39 0.4803845  
## 4 all 101 0.7562283  
## 5 all 16 0.2500000  
## 6 all 79 0.5625440

# Scale/center & append  
bos\_reviews$scaled\_polarity <- scale(bos\_pol$all$polarity)  
  
# Subset positive comments  
pos\_comments <- subset(bos\_reviews$comments, bos\_reviews$scaled\_polarity>0)  
  
# Subset negative comments  
neg\_comments <- subset(bos\_reviews$comments, bos\_reviews$scaled\_polarity<0)  
  
# Paste and collapse the positive comments  
pos\_terms <- paste(pos\_comments, collapse = " ")  
  
# Paste and collapse the negative comments  
neg\_terms <- paste(neg\_comments, collapse = " ")  
  
# Organize  
all\_terms<- c(pos\_terms, neg\_terms)  
  
# VCorpus  
all\_corpus <- VCorpus(VectorSource(all\_terms))  
  
# TDM  
all\_tdm <- TermDocumentMatrix(  
 all\_corpus,   
 control = list(  
 weighting = weightTfIdf,   
 removePunctuation = TRUE,   
 stopwords = stopwords(kind = "en")  
 )  
)  
  
# Column names  
all\_tdm\_m <- as.matrix(all\_tdm)  
colnames(all\_tdm\_m) <- c("positive", "negative")  
  
# Comparison cloud  
comparison.cloud(  
 all\_tdm\_m,   
 max.words = 100,  
 colors = c("darkgreen", "darkred")  
)

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : wellequipped could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : importantly could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : suggested could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : victorian could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : dishes could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : initial could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : response could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : terrible could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : amazingly could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : appliances could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : caring could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : danielle could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : gerald could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : luxurious could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : matter could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : meetings could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : mentioned could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : offers could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : particularly could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : phyllis could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : round could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : saras could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : spaces could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : unique could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : andree could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : broken could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : elevator could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : narrow could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : smaller could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : steep could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : supplied could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : utensils could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : accomodations could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : arboretum could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : baking could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : barry could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : carol could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : case could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : chinese could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : comes could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : continental could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : coupon could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : cramped could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : cynthia could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : dogs could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : fascinating could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : flatscreen could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : flowers could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : homemade could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : hospital could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : inexpensive could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : jairs could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : jannah could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : jims could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : kevin could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : kitty could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : knowledge could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : lily could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : lived could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : luxury could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : max could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : messaged could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : mine could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : mother could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : navigate could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : recommand could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : respectful could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : suite could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : superclean could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : timely could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : walgreens could not be fit on page. It will not be  
## plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : warmly could not be fit on page. It will not be plotted.

## Warning in comparison.cloud(all\_tdm\_m, max.words = 100, colors =  
## c("darkgreen", : warmth could not be fit on page. It will not be plotted.

