Assignment 13

Allison Tessman

2024-04-09

Question 1. Working with a text dataset containing Amazon reviews for an iPhone model. Do the following:

* Create a list of all possible pairs of words (not only adjacent words) that do not contain stopwords and plot the most frequent pairs.

library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.3 ✔ readr 2.1.4  
## ✔ forcats 1.0.0 ✔ stringr 1.5.0  
## ✔ ggplot2 3.5.0 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.2 ✔ tidyr 1.3.0  
## ✔ purrr 1.0.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(ggplot2)  
library(tidytext)  
df <- read\_csv("https://bryantstats.github.io/math475/assignments/Amazon\_Unlocked\_Mobile2.csv")

## Rows: 1451 Columns: 2  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (2): Product\_Name, texts  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

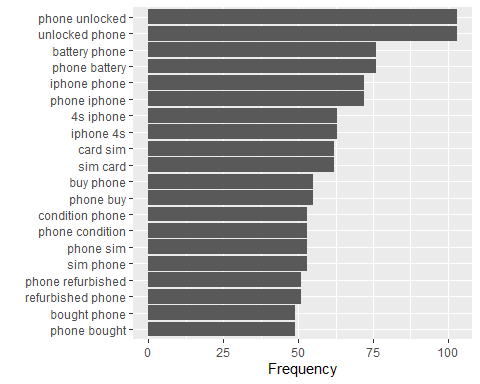
df %>%   
 group\_by(Product\_Name) %>%   
 count(sort = TRUE)

## # A tibble: 1 × 2  
## # Groups: Product\_Name [1]  
## Product\_Name n  
## <chr> <int>  
## 1 Apple iPhone 4s 8GB Unlocked Smartphone w/ 8MP Camera, White (Certified… 1451

df\_section\_words = df %>%   
 mutate(section = c(1:nrow(df))) %>%   
 unnest\_tokens(output = word, input = texts) %>%   
 filter(!word %in% stop\_words$word)  
   
  
library(widyr)  
  
# count words co-occuring within sections  
word\_pairs <- df\_section\_words %>%  
 pairwise\_count(word, section, sort = TRUE)  
  
word\_pairs

## # A tibble: 78,808 × 3  
## item1 item2 n  
## <chr> <chr> <dbl>  
## 1 unlocked phone 103  
## 2 phone unlocked 103  
## 3 battery phone 76  
## 4 phone battery 76  
## 5 iphone phone 72  
## 6 phone iphone 72  
## 7 4s iphone 63  
## 8 iphone 4s 63  
## 9 card sim 62  
## 10 sim card 62  
## # ℹ 78,798 more rows

word\_pairs %>%   
 head(20) %>%   
 mutate(item = paste0(item1, " ", item2)) %>%   
 ggplot() +   
 geom\_col(aes(y = fct\_rev(fct\_reorder(item, -n)), x = n))+  
 labs(y = "", x = 'Frequency')



* Plot the graph of the pairs of words

library(igraph)

##   
## Attaching package: 'igraph'

## The following objects are masked from 'package:lubridate':  
##   
## %--%, union

## The following objects are masked from 'package:dplyr':  
##   
## as\_data\_frame, groups, union

## The following objects are masked from 'package:purrr':  
##   
## compose, simplify

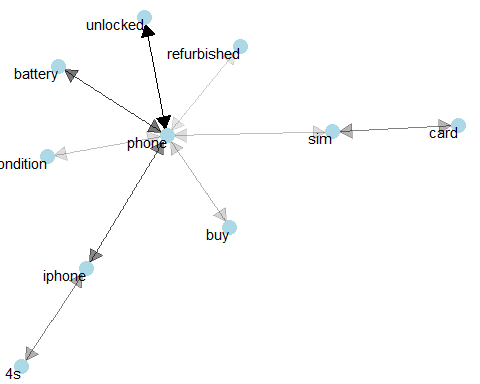
## The following object is masked from 'package:tidyr':  
##   
## crossing

## The following object is masked from 'package:tibble':  
##   
## as\_data\_frame

## The following objects are masked from 'package:stats':  
##   
## decompose, spectrum

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

pairs\_graph <- word\_pairs %>%  
 filter(n > 50) %>%  
 graph\_from\_data\_frame()  
  
library(ggraph)  
set.seed(2024)  
  
a <- grid::arrow(type = "closed", length = unit(.15, "inches"))  
  
ggraph(pairs\_graph, layout = "fr") +  
 geom\_edge\_link(aes(edge\_alpha = n), show.legend = FALSE,  
 arrow = a, end\_cap = circle(.07, 'inches')) +  
 geom\_node\_point(color = "lightblue", size = 5) +  
 geom\_node\_text(aes(label = name), vjust = 1, hjust = 1) +  
 theme\_void()

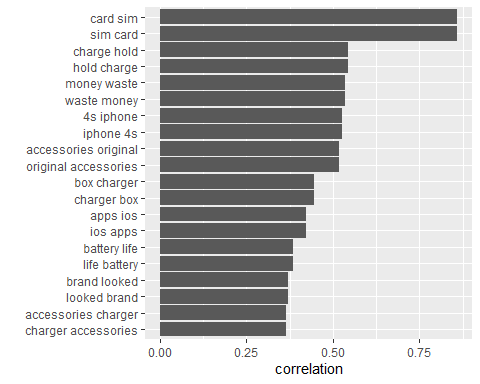


* Compute the correlations of all the pairs and plot the bar chart of the pairs with the highest correlation

word\_cors <- df\_section\_words %>%  
 group\_by(word) %>%  
 filter(n() >= 20) %>%  
 pairwise\_cor(item = word, feature = section, sort = TRUE)  
  
word\_cors

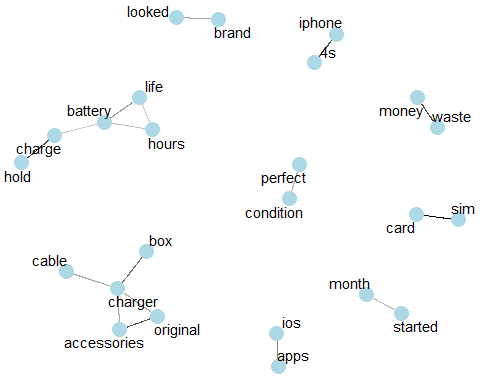
## # A tibble: 8,372 × 3  
## item1 item2 correlation  
## <chr> <chr> <dbl>  
## 1 card sim 0.860  
## 2 sim card 0.860  
## 3 charge hold 0.545  
## 4 hold charge 0.545  
## 5 money waste 0.534  
## 6 waste money 0.534  
## 7 4s iphone 0.526  
## 8 iphone 4s 0.526  
## 9 accessories original 0.518  
## 10 original accessories 0.518  
## # ℹ 8,362 more rows

word\_cors %>%   
 head(20) %>%   
 mutate(item = paste0(item1, " ", item2)) %>%   
 ggplot()+  
 geom\_col(aes(y = fct\_rev(fct\_reorder(item, -correlation)), x = correlation))+  
 labs(y ='')



* Plot the graph of the correlation

library(ggraph)  
library(igraph)  
set.seed(2024)  
  
word\_cors %>%  
 filter(correlation > .3) %>%  
 graph\_from\_data\_frame() %>%  
 ggraph(layout = "fr") +  
 geom\_edge\_link(aes(edge\_alpha = correlation), show.legend = FALSE) +  
 geom\_node\_point(color = "lightblue", size = 5) +  
 geom\_node\_text(aes(label = name), repel = TRUE) +  
 theme\_void()



Question 2. Do Question 1 on your own text dataset.

library(tidyverse)  
library(ggplot2)  
library(tidytext)  
  
df <- read\_csv('~/Applied Analystics SAS Prog/mymath475/CNNtext.csv')

## Rows: 11490 Columns: 3  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (3): id, article, highlights  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

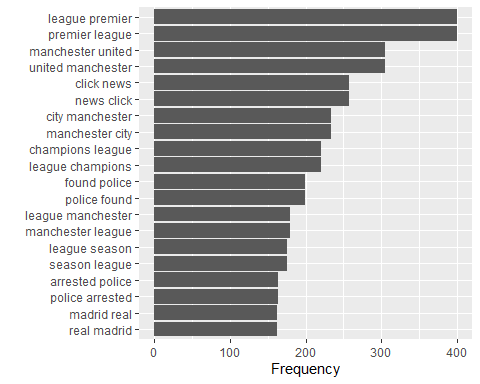
df %>%   
 group\_by(id) %>%   
 count(sort = TRUE)

## # A tibble: 11,490 × 2  
## # Groups: id [11,490]  
## id n  
## <chr> <int>  
## 1 000571afe702684d90c1d222ce70b1e1375c1016 1  
## 2 000642916e3a6c33411c617cf2f3c134a206fba8 1  
## 3 00110802bc6eae0e8e4d3d22e27f458f41be2b22 1  
## 4 00119229166ae09a6ef25c0e10b101ef9eb9cca3 1  
## 5 0013aa16650fbcfbe6edb16ac614ad174cb5d1cf 1  
## 6 00180b7ce54794a52766d795506a94071f7c055b 1  
## 7 001ebaa80dca4a65adf2178b132113cb9e3d5431 1  
## 8 00200e794fa41d3f7ce92cbf43e9fd4cd652bb09 1  
## 9 00217448b38d81a23db66ac362bee25056f58fab 1  
## 10 0021fe8d65bd0d6d76d5fefba2ac02f0c48a43f4 1  
## # ℹ 11,480 more rows

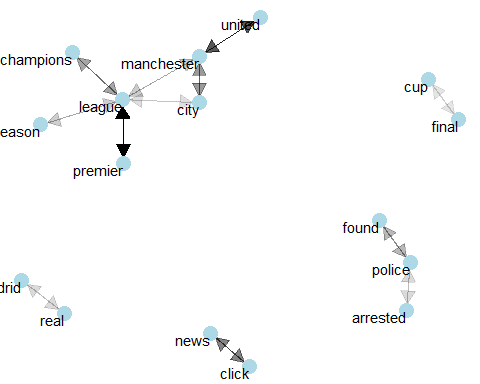
df\_section\_words = df %>%   
 mutate(section = c(1:nrow(df))) %>%   
 unnest\_tokens(output = word, input = highlights) %>%   
 filter(!word %in% stop\_words$word)  
   
  
library(widyr)  
  
# count words co-occuring within sections  
word\_pairs <- df\_section\_words %>%  
 pairwise\_count(word, section, sort = TRUE)  
  
word\_pairs

## # A tibble: 6,089,844 × 3  
## item1 item2 n  
## <chr> <chr> <dbl>  
## 1 league premier 400  
## 2 premier league 400  
## 3 united manchester 305  
## 4 manchester united 305  
## 5 news click 257  
## 6 click news 257  
## 7 manchester city 233  
## 8 city manchester 233  
## 9 champions league 220  
## 10 league champions 220  
## # ℹ 6,089,834 more rows

word\_pairs %>%   
 head(20) %>%   
 mutate(item = paste0(item1, " ", item2)) %>%   
 ggplot() +   
 geom\_col(aes(y = fct\_rev(fct\_reorder(item, -n)), x = n))+  
 labs(y = "", x = 'Frequency')



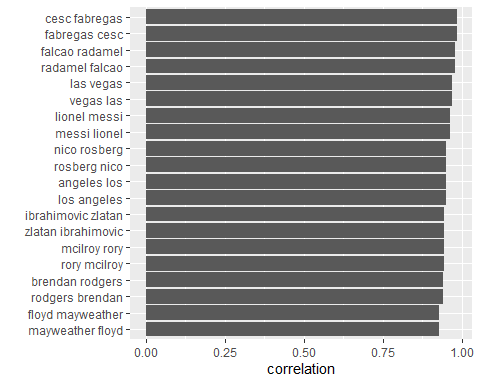
library(igraph)  
pairs\_graph <- word\_pairs %>%  
 filter(n > 150) %>%  
 graph\_from\_data\_frame()  
  
library(ggraph)  
set.seed(2024)  
  
a <- grid::arrow(type = "closed", length = unit(.15, "inches"))  
  
ggraph(pairs\_graph, layout = "fr") +  
 geom\_edge\_link(aes(edge\_alpha = n), show.legend = FALSE,  
 arrow = a, end\_cap = circle(.07, 'inches')) +  
 geom\_node\_point(color = "lightblue", size = 5) +  
 geom\_node\_text(aes(label = name), vjust = 1, hjust = 1) +  
 theme\_void()



word\_cors <- df\_section\_words %>%  
 group\_by(word) %>%  
 filter(n() >= 20) %>%  
 pairwise\_cor(item = word, feature = section, sort = TRUE)  
  
word\_cors

## # A tibble: 10,352,306 × 3  
## item1 item2 correlation  
## <chr> <chr> <dbl>  
## 1 fabregas cesc 0.983  
## 2 cesc fabregas 0.983  
## 3 falcao radamel 0.979  
## 4 radamel falcao 0.979  
## 5 vegas las 0.969  
## 6 las vegas 0.969  
## 7 messi lionel 0.961  
## 8 lionel messi 0.961  
## 9 rosberg nico 0.951  
## 10 nico rosberg 0.951  
## # ℹ 10,352,296 more rows

word\_cors %>%   
 head(20) %>%   
 mutate(item = paste0(item1, " ", item2)) %>%   
 ggplot()+  
 geom\_col(aes(y = fct\_rev(fct\_reorder(item, -correlation)), x = correlation))+  
 labs(y ='')



library(ggraph)  
library(igraph)  
set.seed(2024)  
  
word\_cors %>%  
 filter(correlation > .75) %>%  
 graph\_from\_data\_frame() %>%  
 ggraph(layout = "fr") +  
 geom\_edge\_link(aes(edge\_alpha = correlation), show.legend = FALSE) +  
 geom\_node\_point(color = "lightblue", size = 5) +  
 geom\_node\_text(aes(label = name), repel = TRUE) +  
 theme\_void()

