text_book_ch2

Michael Robinson

2023-11-09

Introduction:

In this lab I will use the code from the reading, to examine Text mining, using three lexicons (bing, nrc, and afinn), I will then use an additional lexicon (loughran) to perform further analysis. I will then create a second R chunk using a different corpus and all four lexicons.

References

R for Data Science by Hadley Wickham & Garrett Grolemund (2017). Package `tidytext`. Retrieved from https://www.tidytextmining.com/

Silge, Julia, PhD. & Robinson, David, PhD. (2017). Text Mining with R: A Tidy Approach. O'Reilly Media, Inc.

```
library(tidytext)
library(janeaustenr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

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

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

```
library(stringr)
library(tidyr)
library(ggplot2)
library(wordcloud)
```

```
## Loading required package: RColorBrewer
```

```r
library(lexicon)
library(textdata)



text_df <- read.csv("/Users/michaelrobinson/Data_607/tweets_data.csv", stringsAsFactors = FALSE, header

get_sentiments("afinn")
```

```
## # A tibble: 2,477 x 2
##    word       value
##    <chr>      <dbl>
##  1 abandon       -2
##  2 abandoned     -2
##  3 abandons      -2
##  4 abducted      -2
##  5 abduction     -2
##  6 abductions    -2
##  7 abhor         -3
##  8 abhorred      -3
##  9 abhorrent     -3
## 10 abhors        -3
## # i 2,467 more rows
```

```r
get_sentiments("bing")
```

```
## # A tibble: 6,786 x 2
##    word        sentiment
##    <chr>       <chr>
##  1 2-faces     negative
##  2 abnormal    negative
##  3 abolish     negative
##  4 abominable  negative
##  5 abominably  negative
##  6 abominate   negative
##  7 abomination negative
##  8 abort       negative
##  9 aborted     negative
## 10 aborts      negative
## # i 6,776 more rows
```

```r
get_sentiments("nrc")
```

```
## # A tibble: 13,872 x 2
##    word        sentiment
##    <chr>       <chr>
##  1 abacus      trust
##  2 abandon     fear
##  3 abandon     negative
##  4 abandon     sadness
##  5 abandoned   anger
##  6 abandoned   fear
##  7 abandoned   negative
##  8 abandoned   sadness
##  9 abandonment anger
## 10 abandonment fear
## # i 13,862 more rows
```

```
get_sentiments("loughran")
```

```
## # A tibble: 4,150 x 2
##    word         sentiment
##    <chr>        <chr>
##  1 abandon      negative
##  2 abandoned    negative
##  3 abandoning   negative
##  4 abandonment  negative
##  5 abandonments negative
##  6 abandons     negative
##  7 abdicated    negative
##  8 abdicates    negative
##  9 abdicating   negative
## 10 abdication   negative
## # i 4,140 more rows
```

```
tidy_books <- austen_books() %>%
  group_by(book) %>%
  mutate(
    linenumber = row_number(),
    chapter = cumsum(str_detect(text,
                                regex("^chapter [\\divxlc]",
                                      ignore_case = TRUE)))) %>%
  ungroup() %>%
  unnest_tokens(word, text)

nrc_joy <- get_sentiments("nrc") %>%
  filter(sentiment == "joy")

tidy_books %>%
  filter(book == "Emma") %>%
  inner_join(nrc_joy) %>%
  count(word, sort = TRUE)
```

```
## Joining with ‘by = join_by(word)‘
```

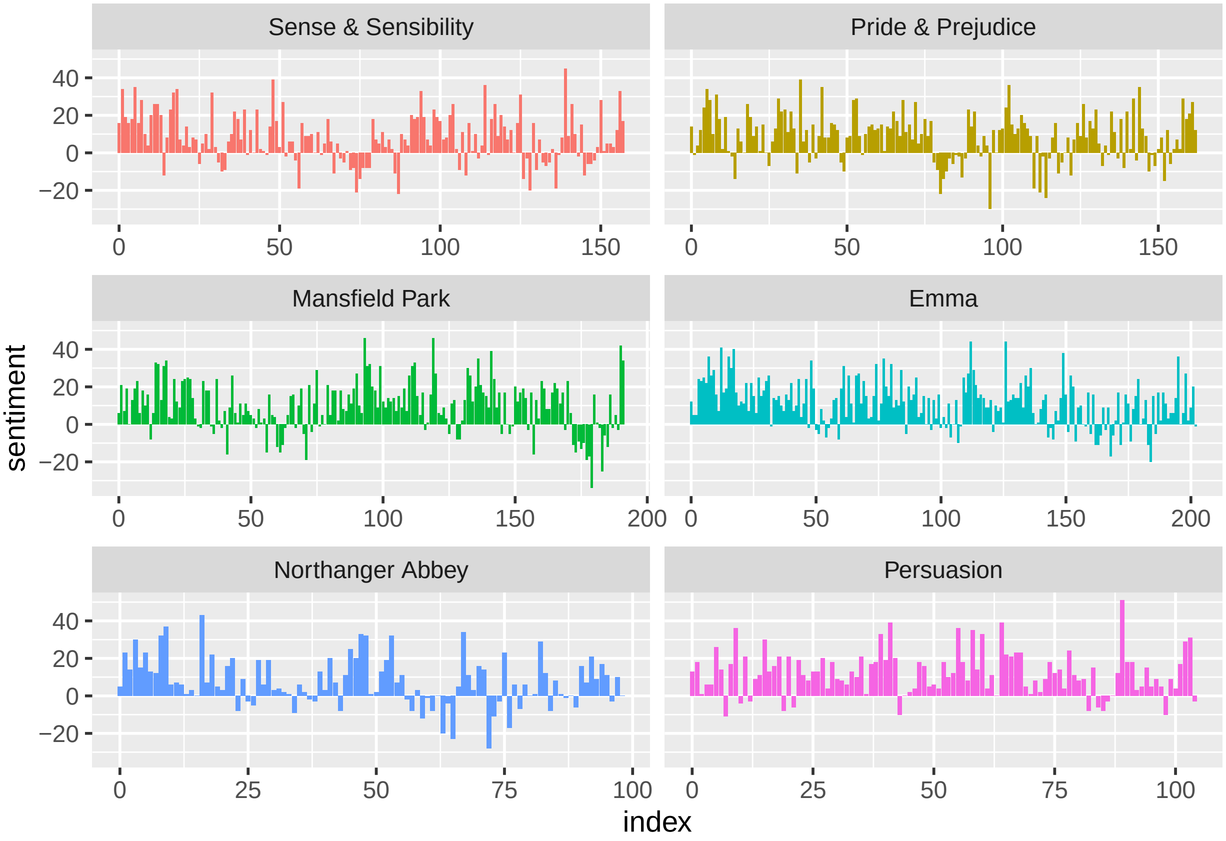
```
## # A tibble: 301 x 2
##    word        n
##    <chr>   <int>
##  1 good      359
##  2 friend    166
##  3 hope      143
##  4 happy     125
##  5 love      117
##  6 deal       92
##  7 found        92
##  8 present      89
##  9 kind         82
## 10 happiness    76
## # i 291 more rows
```

```
jane_austen_sentiment <- tidy_books %>%
  inner_join(get_sentiments("bing")) %>%
  count(book, index = linenumber %/% 80, sentiment) %>%
  pivot_wider(names_from = sentiment, values_from = n, values_fill = 0) %>%
  mutate(sentiment = positive - negative)
```

```
## Joining with ‘by = join_by(word)‘
```

```
## Warning in inner_join(., get_sentiments("bing")): Detected an unexpected many-to-many relationship be
## i Row 435434 of ‘x‘ matches multiple rows in ‘y‘.
## i Row 5051 of ‘y‘ matches multiple rows in ‘x‘.
## i If a many-to-many relationship is expected, set ‘relationship =
##   "many-to-many"‘ to silence this warning.
```

```
ggplot(jane_austen_sentiment, aes(index, sentiment, fill = book)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~book, ncol = 2, scales = "free_x")
```



```r
pride_prejudice <- tidy_books %>%
  filter(book == "Pride & Prejudice")

afinn <- pride_prejudice %>%
  inner_join(get_sentiments("afinn")) %>%
  group_by(index = linenumber %/% 80) %>%
  summarise(sentiment = sum(value)) %>%
  mutate(method = "AFINN")
```

```
## Joining with ‘by = join_by(word)‘
```

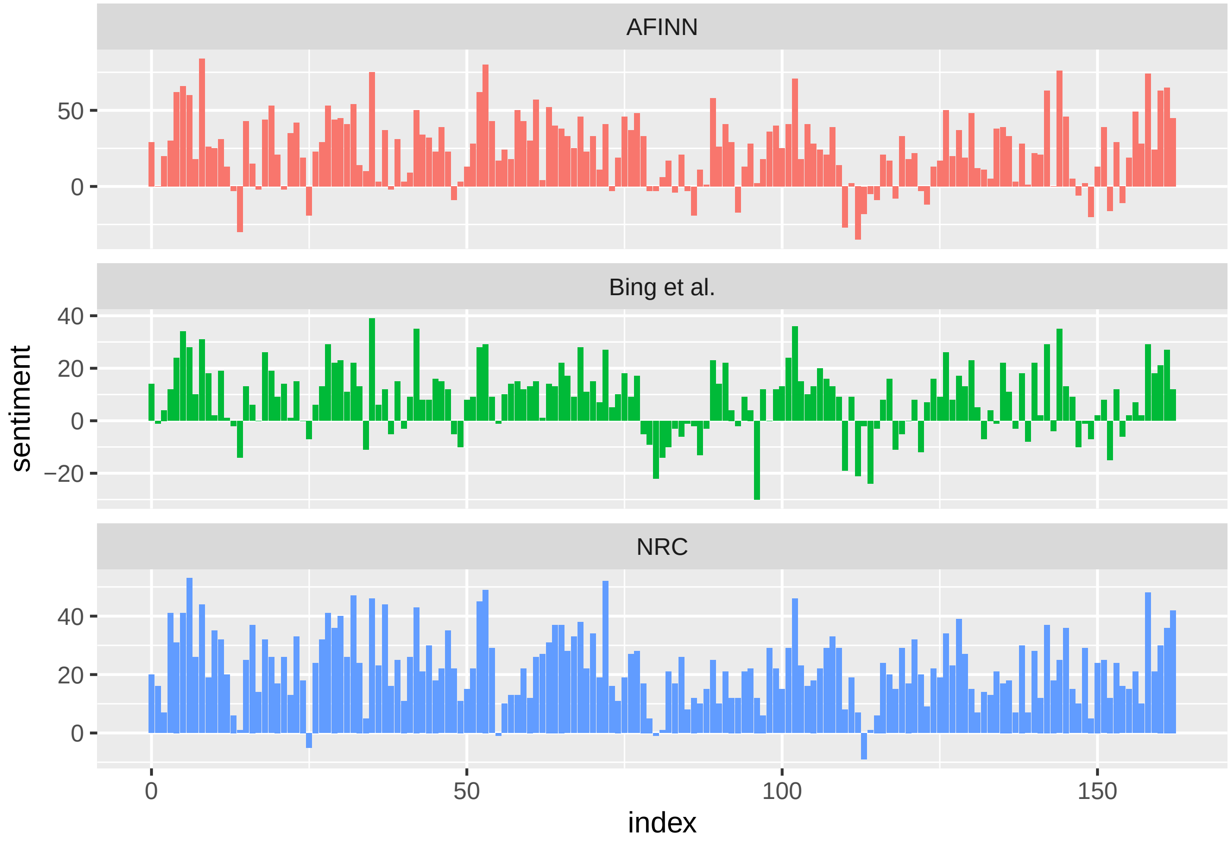
```r
bing_and_nrc <- bind_rows(
  pride_prejudice %>%
    inner_join(get_sentiments("bing")) %>%
    mutate(method = "Bing et al."),
  pride_prejudice %>%
    inner_join(get_sentiments("nrc") %>%
                 filter(sentiment %in% c("positive",
                                         "negative"))
    ) %>%
    mutate(method = "NRC")) %>%
  count(method, index = linenumber %/% 80, sentiment) %>%
  pivot_wider(names_from = sentiment,
              values_from = n,
              values_fill = 0) %>%
  mutate(sentiment = positive - negative)
```

```
## Joining with ‘by = join_by(word)‘
```

```
## Joining with ‘by = join_by(word)‘
```

```
## Warning in inner_join(., get_sentiments("nrc") %>% filter(sentiment %in% : Detected an unexpected ma
## i Row 215 of ‘x‘ matches multiple rows in ‘y‘.
## i Row 5178 of ‘y‘ matches multiple rows in ‘x‘.
## i If a many-to-many relationship is expected, set ‘relationship =
##   "many-to-many"‘ to silence this warning.
```

```r
bind_rows(afinn,
          bing_and_nrc) %>%
  ggplot(aes(index, sentiment, fill = method)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~method, ncol = 1, scales = "free_y")
```

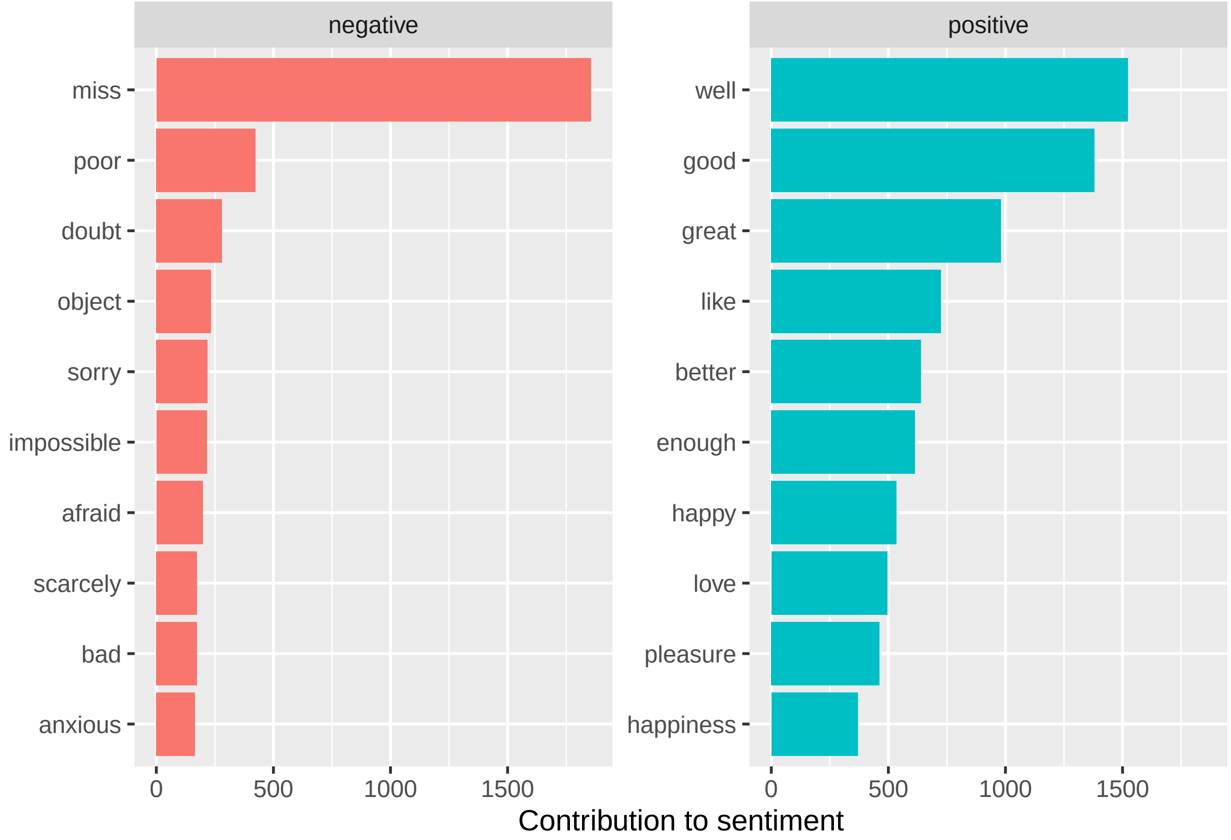


```
bing_word_counts <- tidy_books %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE) %>%
  ungroup()
```

```
## Joining with ‘by = join_by(word)‘
```

```
## Warning in inner_join(., get_sentiments("bing")): Detected an unexpected many-to-many relationship be
## i Row 435434 of ‘x‘ matches multiple rows in ‘y‘.
## i Row 5051 of ‘y‘ matches multiple rows in ‘x‘.
## i If a many-to-many relationship is expected, set ‘relationship =
##   "many-to-many"‘ to silence this warning.
```

```
bing_word_counts %>%
  group_by(sentiment) %>%
  slice_max(n, n = 10) %>%
  ungroup() %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(n, word, fill = sentiment)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~sentiment, scales = "free_y") +
  labs(x = "Contribution to sentiment",
       y = NULL)
```



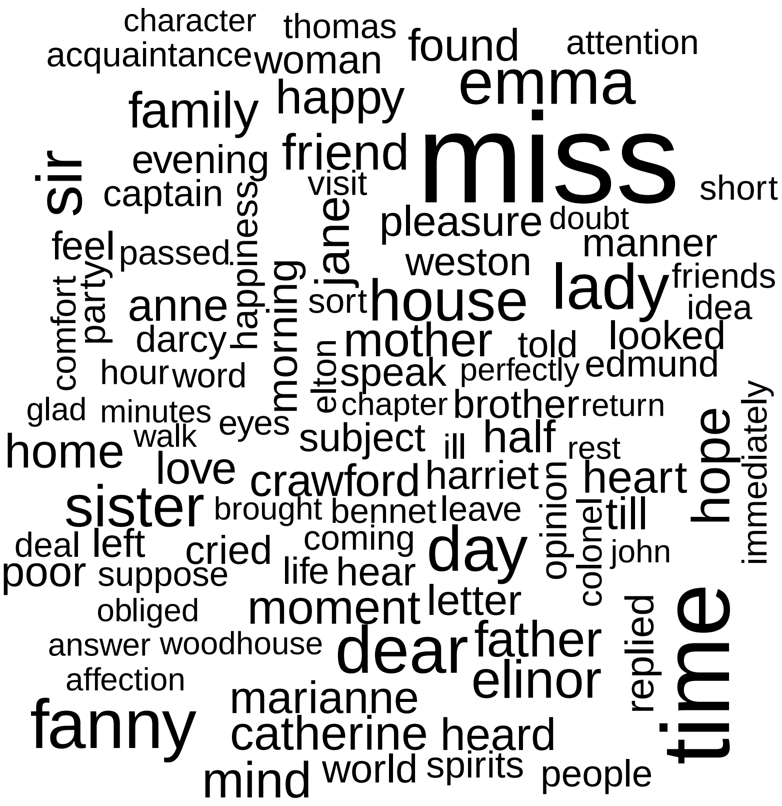
```
custom_stop_words <- bind_rows(tibble(word = c("miss"),
                                      lexicon = c("custom")),
                               stop_words)

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

```
## Joining with `by = join_by(word)`
```

```
## Warning in wordcloud(word, n, max.words = 100): elizabeth could not be fit on
## page. It will not be plotted.
```

```
## Warning in wordcloud(word, n, max.words = 100): feelings could not be fit on
## page. It will not be plotted.
```

```
## Warning in wordcloud(word, n, max.words = 100): knightley could not be fit on
## page. It will not be plotted.
```



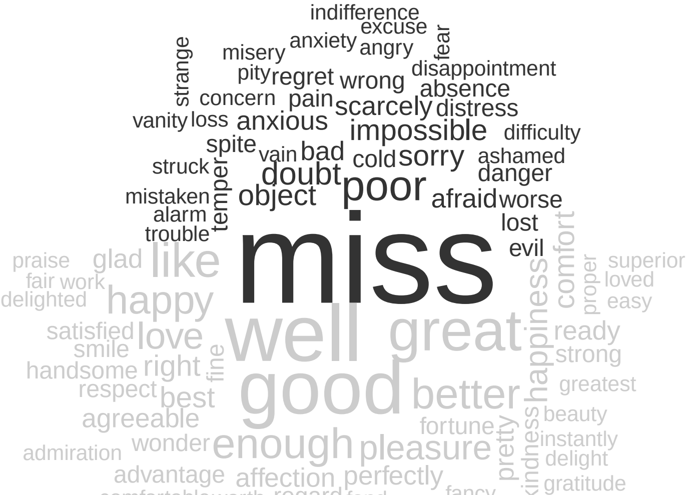
```r
library(reshape2)
```

```
##
## Attaching package: 'reshape2'
##
## The following object is masked from 'package:tidyr':
##
##     smiths
```

```r
tidy_books %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE) %>%
  acast(word ~ sentiment, value.var = "n", fill = 0) %>%
  comparison.cloud(colors = c("gray20", "gray80"),
                   max.words = 100)
```

```
## Joining with `by = join_by(word)`
```

```
## Warning in inner_join(., get_sentiments("bing")): Detected an unexpected many-to-many relationship be
## i Row 435434 of `x` matches multiple rows in `y`.
## i Row 5051 of `y` matches multiple rows in `x`.
## i If a many-to-many relationship is expected, set `relationship =
##   "many-to-many"` to silence this warning.
```



```r
p_and_p_sentences <- tibble(text = prideprejudice) %>%
  unnest_tokens(sentence, text, token = "sentences")
p_and_p_sentences$sentence[2]
```

```
## [1] "by jane austen"
```

```r
austen_chapters <- austen_books() %>%
  group_by(book) %>%
  unnest_tokens(chapter, text, token = "regex",
                pattern = "Chapter|CHAPTER [\\dIVXLC]") %>%
  ungroup()

austen_chapters %>%
  group_by(book) %>%
  summarise(chapters = n())
```

```
## # A tibble: 6 x 2
##   book                chapters
##   <fct>                  <int>
## 1 Sense & Sensibility       51
## 2 Pride & Prejudice         62
## 3 Mansfield Park            49
## 4 Emma                      56
## 5 Northanger Abbey          32
## 6 Persuasion                25
```

```
bingnegative <- get_sentiments("bing") %>%
  filter(sentiment == "negative")

wordcounts <- tidy_books %>%
  group_by(book, chapter) %>%
  summarize(words = n())
```

```
## ‘summarise()‘ has grouped output by ’book’. You can override using the
## ‘.groups‘ argument.
```

```
tidy_books %>%
  semi_join(bingnegative) %>%
  group_by(book, chapter) %>%
  summarize(negativewords = n()) %>%
  left_join(wordcounts, by = c("book", "chapter")) %>%
  mutate(ratio = negativewords/words) %>%
  filter(chapter != 0) %>%
  slice_max(ratio, n = 1) %>%
  ungroup()
```

```
## Joining with ‘by = join_by(word)‘
## ‘summarise()‘ has grouped output by ’book’. You can override using the
## ‘.groups‘ argument.
```

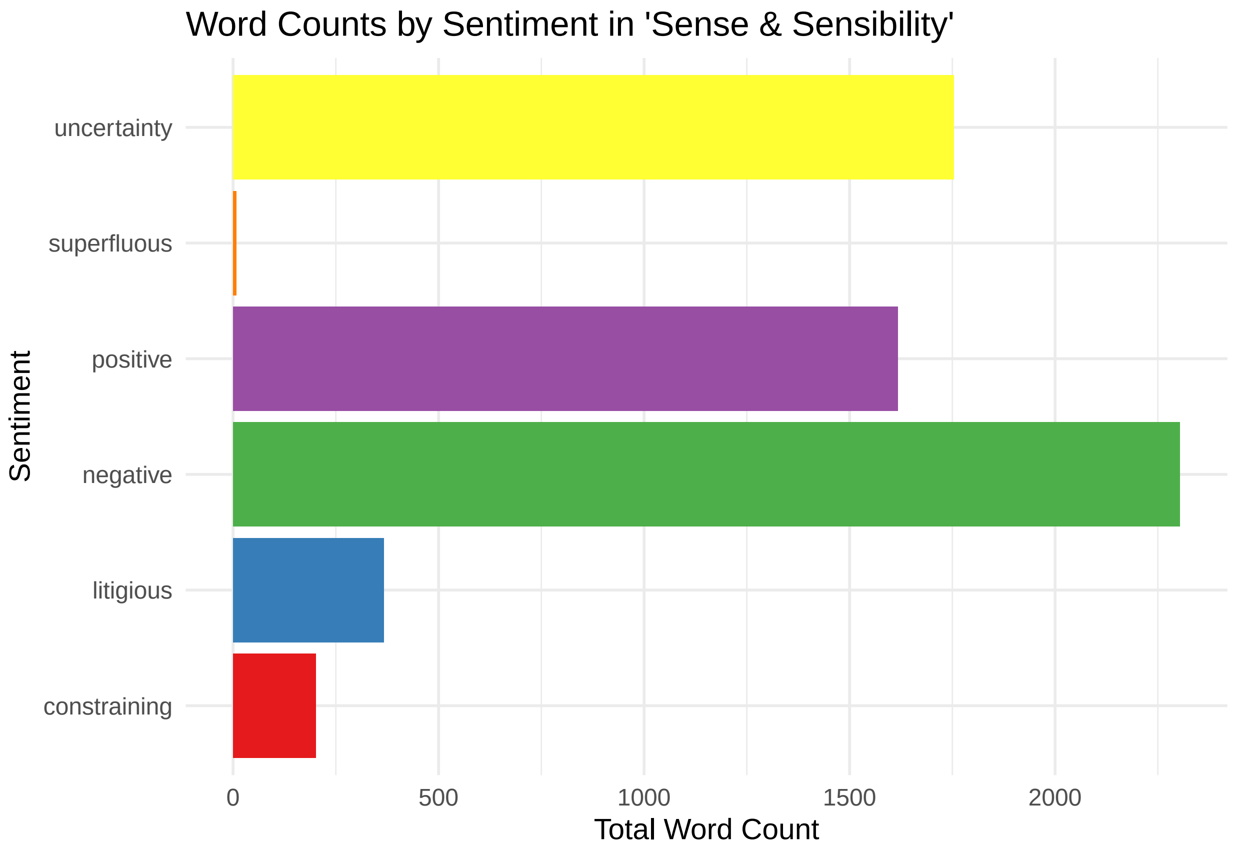
```
## # A tibble: 6 x 5
##   book                chapter negativewords words  ratio
##   <fct>                 <int>         <int> <int>  <dbl>
## 1 Sense & Sensibility      43           161  3405 0.0473
## 2 Pride & Prejudice        34           111  2104 0.0528
## 3 Mansfield Park           46           173  3685 0.0469
## 4 Emma                     15           151  3340 0.0452
## 5 Northanger Abbey         21           149  2982 0.0500
## 6 Persuasion                4            62  1807 0.0343
```

```
loughran_lexicon <- get_sentiments("loughran")

loughran_sentiment <- tidy_books %>%
  filter(book == "Sense & Sensibility") %>%
  inner_join(loughran_lexicon, by = c(word = "word")) %>%
  count(word, sentiment, sort = TRUE)
```

```
## Warning in inner_join(., loughran_lexicon, by = c(word = "word")): Detected an unexpected many-to-man
## i Row 1252 of ‘x‘ matches multiple rows in ‘y‘.
## i Row 2772 of ‘y‘ matches multiple rows in ‘x‘.
## i If a many-to-many relationship is expected, set ‘relationship =
##   "many-to-many"‘ to silence this warning.
```

```
loughran_summary <- loughran_sentiment %>%
  group_by(sentiment) %>%
  summarise(total_count = sum(n)) %>%
  ungroup()
```

```r
#create a bar plot
ggplot(loughran_summary, aes(x = sentiment, y = total_count, fill = sentiment)) +
  geom_bar(stat = "identity") +
  theme_minimal() +
  labs(title = "Word Counts by Sentiment in 'Sense & Sensibility'",
       x = "Sentiment",
       y = "Total Word Count") +
  scale_fill_brewer(palette = "Set1") +
  theme(legend.position = "none") + coord_flip()
```



Introduction

In this chunk of the assignment I will use a pdf version on the book A Journey to the center of the earth. I will load the pdf file, then create a corpus and do some text processing. I will then use the lexicons (AFINN,Bing,nrc and loughran) to do analysis on the book and create some visualization.

```r
library(pdftools)
```

```
## Using poppler version 23.04.0
```

```r
library(tm)
```

```
## Loading required package: NLP
```

```
##
## Attaching package: ’NLP’

## The following object is masked from ’package:ggplot2’:
##
##     annotate
```

```
library(tidytext)
library(dplyr)
library(ggplot2)
library(textdata)
library(RefManageR)

# Reference:

bib <- BibEntry(
  bibtype = "Book",
  title = "A Journey to the center of the Earth",
  author = "Jules Verne",
  translator = "Fredrick Amadeus Malleson",
  year = "1871",
  publisher = "Griffith and Farran",
  address = "England"
)

#print(bib)

Book <- "A-Journey-to-the-Centre-of-the-Earth.pdf"

# Read the text from the PDF
journey_cent <- pdf_text(Book)

# Create corpus
document <- Corpus(VectorSource(journey_cent))

# Text preprocessing
document <- tm_map(document, content_transformer(tolower))
```

```
## Warning in tm_map.SimpleCorpus(document, content_transformer(tolower)):
## transformation drops documents
```

```
document <- tm_map(document, removeNumbers)
```

```
## Warning in tm_map.SimpleCorpus(document, removeNumbers): transformation drops
## documents
```

```
document <- tm_map(document, removeWords, stopwords("english"))
```

```
## Warning in tm_map.SimpleCorpus(document, removeWords, stopwords("english")):
## transformation drops documents
```

```r
document <- tm_map(document, removePunctuation, preserve_intra_word_dashes = TRUE)
```

```
## Warning in tm_map.SimpleCorpus(document, removePunctuation,
## preserve_intra_word_dashes = TRUE): transformation drops documents
```

```r
document <- tm_map(document, stripWhitespace)
```

```
## Warning in tm_map.SimpleCorpus(document, stripWhitespace): transformation drops
## documents
```

```r
# Create a Document-Term Matrix
Book_Jorney <- DocumentTermMatrix(document)

# Convert the Document-Term Matrix into a tidy format
Book_Jorney_tidy <- tidy(Book_Jorney)
names(Book_Jorney_tidy)[2] <- 'word'

# Access the lexicons
get_sentiments("afinn")
```

```
## # A tibble: 2,477 x 2
##    word       value
##    <chr>      <dbl>
##  1 abandon       -2
##  2 abandoned     -2
##  3 abandons      -2
##  4 abducted      -2
##  5 abduction     -2
##  6 abductions    -2
##  7 abhor         -3
##  8 abhorred      -3
##  9 abhorrent     -3
## 10 abhors        -3
## # i 2,467 more rows
```

```r
get_sentiments("bing")
```

```
## # A tibble: 6,786 x 2
##    word        sentiment
##    <chr>       <chr>
##  1 2-faces     negative
##  2 abnormal    negative
##  3 abolish     negative
##  4 abominable  negative
##  5 abominably  negative
##  6 abominate   negative
##  7 abomination negative
##  8 abort       negative
##  9 aborted     negative
## 10 aborts      negative
## # i 6,776 more rows
```

```
get_sentiments("nrc")
```

```
## # A tibble: 13,872 x 2
##    word        sentiment
##    <chr>       <chr>
##  1 abacus      trust
##  2 abandon     fear
##  3 abandon     negative
##  4 abandon     sadness
##  5 abandoned   anger
##  6 abandoned   fear
##  7 abandoned   negative
##  8 abandoned   sadness
##  9 abandonment anger
## 10 abandonment fear
## # i 13,862 more rows
```

```
# using the Bing lexicon
Book_Jorney_bing <- Book_Jorney_tidy %>%
  inner_join(get_sentiments("bing"), by = c(word = "word"))
```

```
## Warning in inner_join(., get_sentiments("bing"), by = c(word = "word")): Detected an unexpected many-
## i Row 2175 of ‘x‘ matches multiple rows in ‘y‘.
## i Row 2736 of ‘y‘ matches multiple rows in ‘x‘.
## i If a many-to-many relationship is expected, set ‘relationship =
##   "many-to-many"‘ to silence this warning.
```

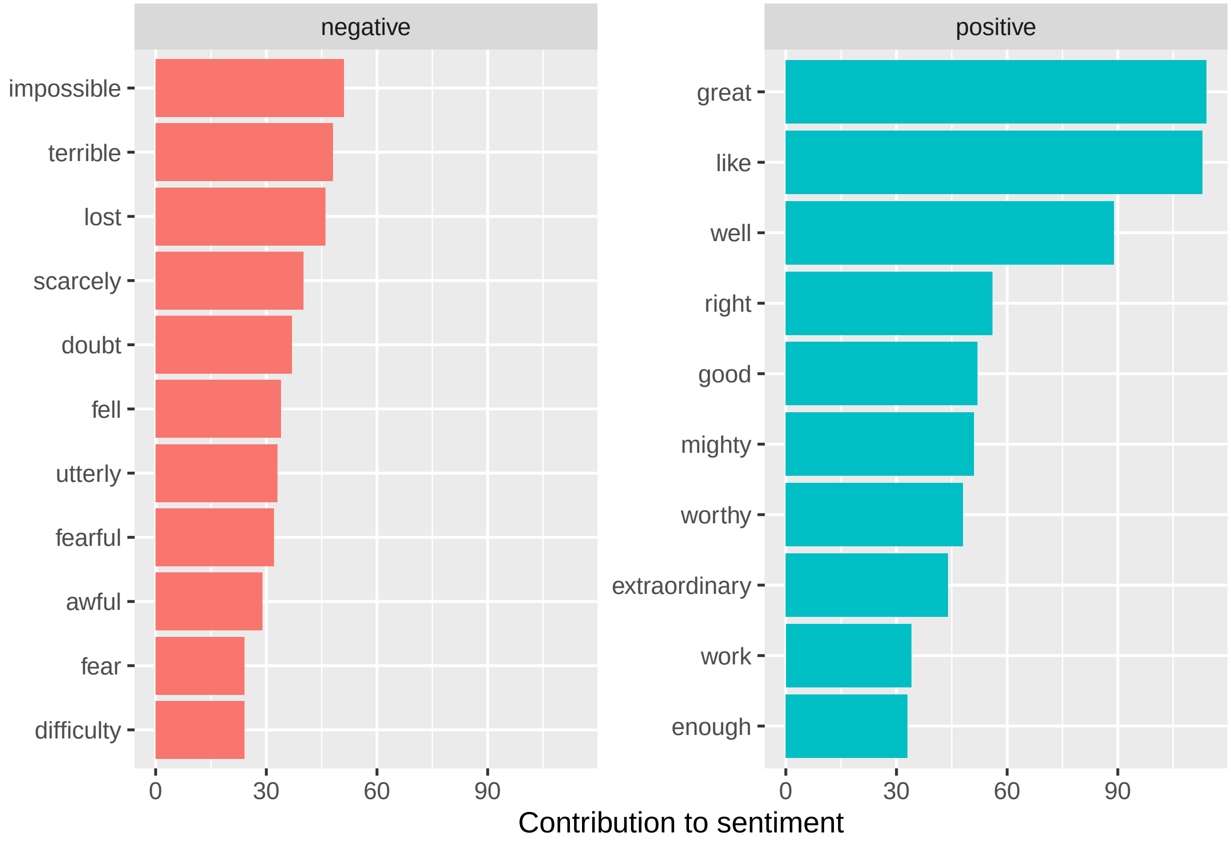
```
# Using the AFINN lexicon
Book_Jorney_afinn <- Book_Jorney_tidy %>%
  inner_join(get_sentiments("afinn"), by = c(word = "word"))

# Filtering the joy words from the NRC lexicon
nrcjoy <- get_sentiments("nrc") %>%
  filter(sentiment == "joy")
Book_Jorney_nrcjoy <- Book_Jorney_tidy %>%
  inner_join(nrcjoy) %>%
  count(word, sort = TRUE)
```

```
## Joining with ‘by = join_by(word)‘
```

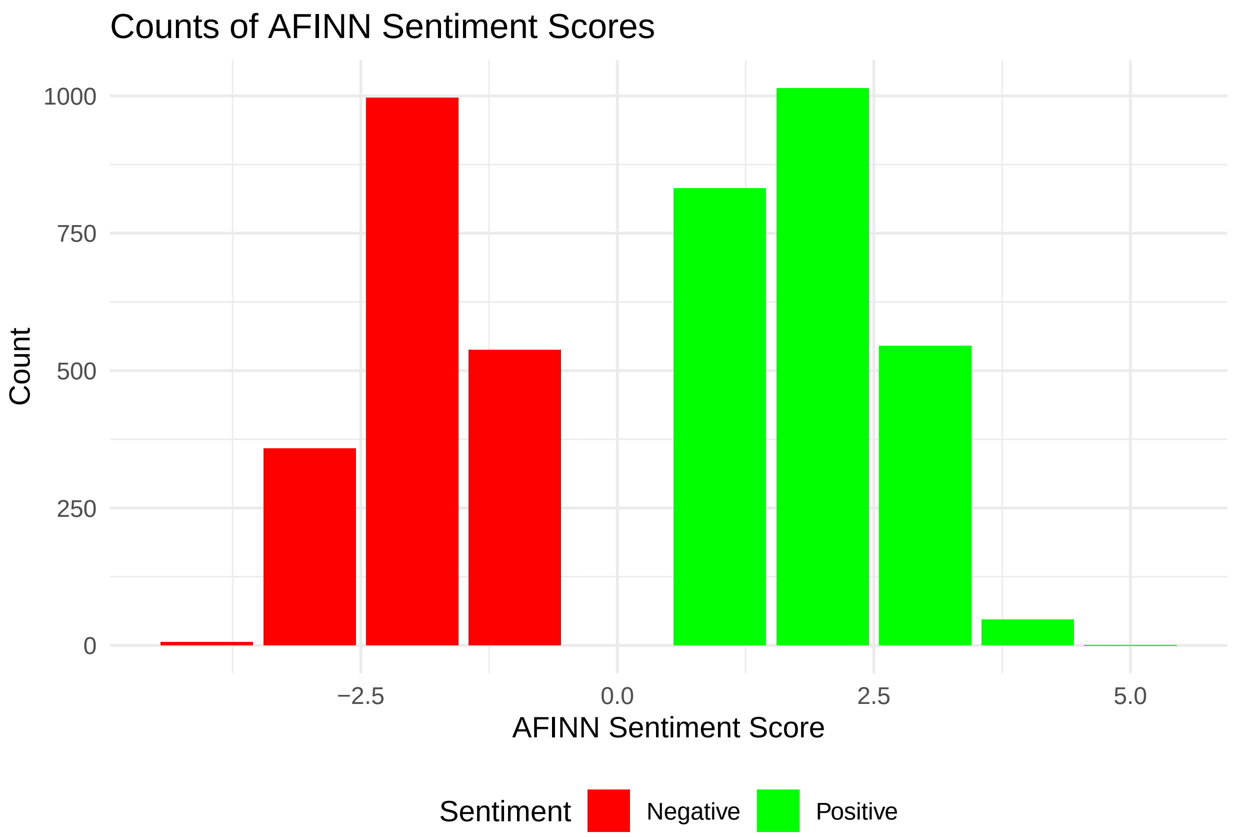
```
# Filtering the fear words from the NRC lexicon
nrcfear <- get_sentiments("nrc") %>%
  filter(sentiment == "fear")
Book_Jorney_nrcfear <- Book_Jorney_tidy %>%
  inner_join(nrcfear) %>%
  count(word, sort = TRUE)
```

```
## Joining with ‘by = join_by(word)‘
```

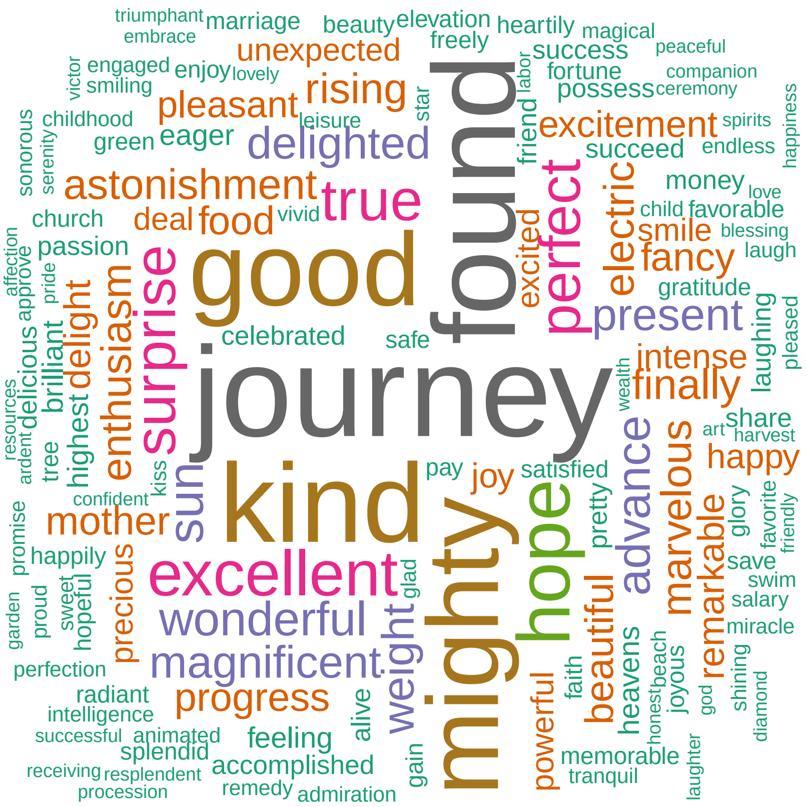
```r
# create a frequency count for the Bing lexicon
Book_Jorney_bing_count <- Book_Jorney_bing %>%
  count(word, sentiment, sort = TRUE)

# AFINN lexicon, sum the scores for each word
Book_Jorney_afinn_sum <- Book_Jorney_afinn %>%
  group_by(word) %>%
  summarize(score_sum = sum(value, na.rm = TRUE)) %>%
  ungroup() %>%
  arrange(desc(score_sum))

# Calculate the count of each sentiment score
Book_Jorney_afinn_count <- Book_Jorney_afinn %>%
  group_by(value) %>%
  summarize(count = n()) %>%
  ungroup() %>%
  arrange(desc(count))

# Calculate the frequency of words that have an AFINN score
Book_Jorney_afinn_frequency <- Book_Jorney_afinn %>%
  count(word, sort = TRUE)

# Bar plot for Bing lexicon
Book_Jorney_bing_count %>%
  group_by(sentiment) %>%
  slice_max(n, n = 10) %>%
  ungroup() %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(n, word, fill = sentiment)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~sentiment, scales = "free_y") +
  labs(x = "Contribution to sentiment",
       y = NULL)
```



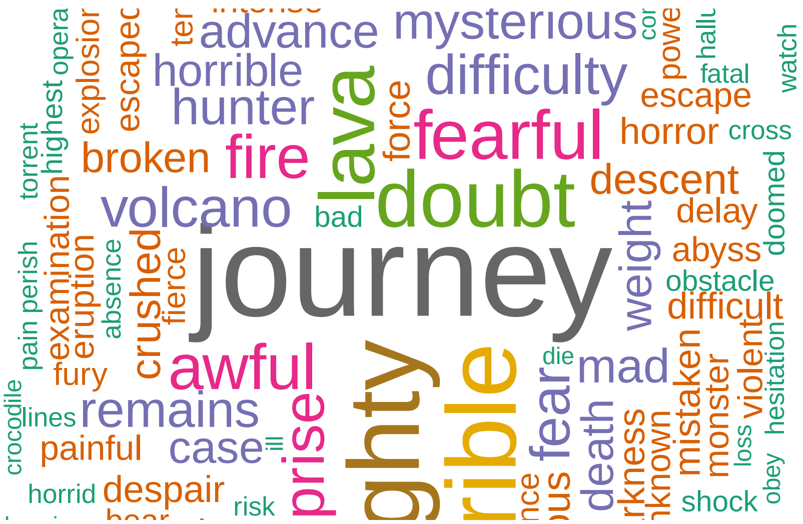
```
# Bar plot for AFINN lexicon
ggplot(Book_Jorney_afinn_count, aes(x = value, y = count)) +
  geom_bar(stat = "identity", aes(fill = value > 0)) +  # Color bars by positive or negative sentiment
  scale_fill_manual(values = c("red", "green"), name = "Sentiment",
                    labels = c("Negative", "Positive")) +
  labs(x = "AFINN Sentiment Score", y = "Count", title = "Counts of AFINN Sentiment Scores") +
  theme_minimal() +
  theme(legend.position = "bottom")
```



```
Book_Jorney_nrcjoy <- Book_Jorney_nrcjoy %>%
  arrange(desc(n))

# Create a wordcloud of nrc joy words

wordcloud(words = Book_Jorney_nrcjoy$word,
          freq = Book_Jorney_nrcjoy$n,
          min.freq = 1,
          max.words = 145,
          random.order = FALSE,
          rot.per = 0.35,
          scale = c(4, 0.5),
          colors = brewer.pal(8, "Dark2"))
```



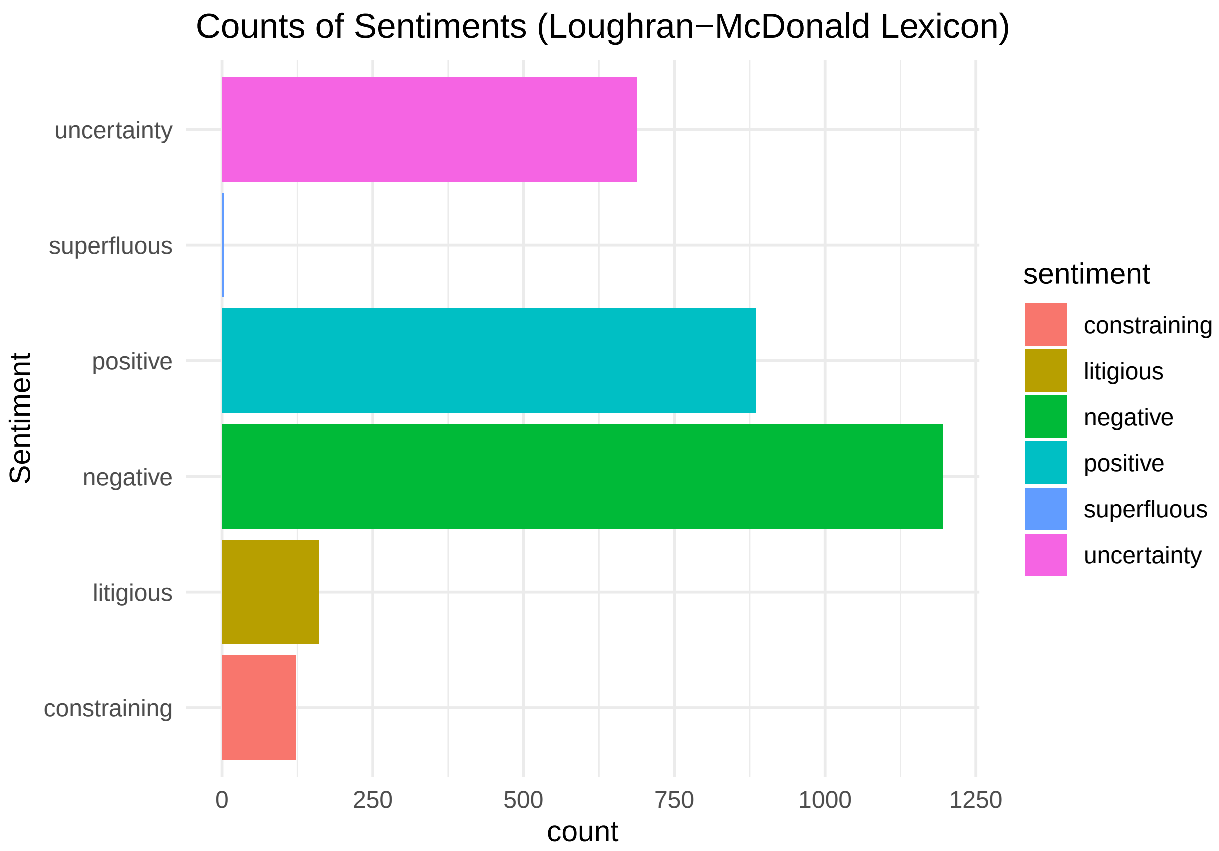
```r
# Create a wordcloud of nrc fear words

Book_Jorney_nrcfear <- Book_Jorney_nrcfear %>%
  arrange(desc(n))

wordcloud(words = Book_Jorney_nrcfear$word,
          freq = Book_Jorney_nrcfear$n,
          min.freq = 1,
          max.words = 110,
          random.order = FALSE,
          rot.per = 0.35,
          scale = c(4, 0.5),
          colors = brewer.pal(8, "Dark2"))
```



```
loughran_lexicon <- get_sentiments("loughran")


Book_Journey_loughran <- Book_Jorney_tidy %>%
  inner_join(loughran_lexicon, by = c(word = "word"))
```

```
## Warning in inner_join(., loughran_lexicon, by = c(word = "word")): Detected an unexpected many-to-mar
## i Row 334 of `x` matches multiple rows in `y`.
## i Row 2356 of `y` matches multiple rows in `x`.
## i If a many-to-many relationship is expected, set `relationship =
##   "many-to-many"` to silence this warning.
```

```
# Count the frequency of each sentiment
Book_Journey_loughran_count <- Book_Journey_loughran %>%
  count(sentiment, sort = TRUE) %>%
  mutate(lexicon = "Loughran-McDonald") # Add a column for the lexicon name


ggplot(Book_Journey_loughran_count, aes(x = sentiment, y = n, fill = sentiment)) +
  geom_bar(stat = "identity") +
  labs(x = "Sentiment", y = "count", title = " Counts of Sentiments (Loughran-McDonald Lexicon)") + ther
```


Counts of Sentiments (Loughran−McDonald Lexicon)
Sentiment
uncertainty
superfluous
positive
negative
litigious
constraining
0
250
500
750
1000
1250
count
sentiment
constraining
litigious
negative
positive
superfluous
uncertainty