DATA 607 Sentiment Analysis

Warner Alexis

2023-11-12

Sentiment Analysis

In this exercise, we will provide some the of code written in chapter 2 to do our analysis. We will load a new corpus based on all the books written by 'Luther, Martin'.

```
# load package
library(janeaustenr)
library(gutenbergr)
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(tidytext)
library(scales)
library(broom)
library(tm)
```

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

```
library(quanteda)
```

```
## Package version: 4.0.0
## Unicode version: 13.0
## ICU version: 69.1
```

```
## Parallel computing: 20 of 20 threads used.
```

```
## See https://quanteda.io for tutorials and examples.
```

```
##
## Attaching package: 'quanteda'
```

```
## The following object is masked from 'package:tm':
##
##     stopwords
```

```
## The following objects are masked from 'package:NLP':
##
##     meta, meta<-
```

```
library(stringr)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats   1.0.0     v readr     2.1.4
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.2
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x ggplot2::annotate() masks NLP::annotate()
## x readr::col_factor() masks scales::col_factor()
## x purrr::discard()    masks scales::discard()
## x dplyr::filter()     masks stats::filter()
## x dplyr::lag()        masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

Example of Codes in Chapter 2

Sentiment analysis provides a way to get understand sentiment expressed in text document. we learn how to tidy a text and create different lexicons.

```
## 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
```

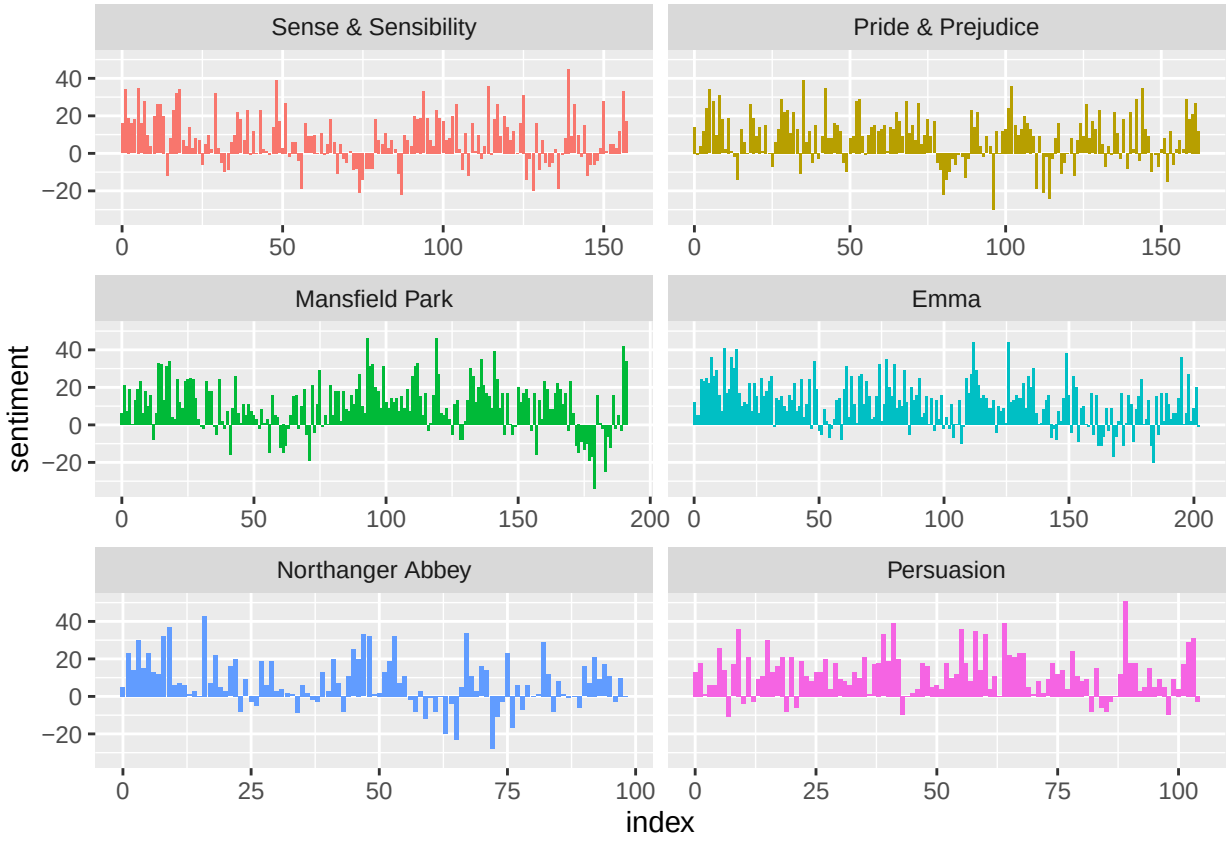
```
# creat the the sentiment dataset usinh inner_join
jane_austen_sentiment <- tidy_books %>%
  inner_join(get_sentiments("bing")) %>%
  count(book, index = linenumber %/% 80, sentiment) %>%
  pivot_wider(names_from = sentiment, values_from = n, values_fill = 0) %>%
  mutate(sentiment = positive - negative)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

```
## 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.
```

```
# plot the result
ggplot(jane_austen_sentiment, aes(index, sentiment, fill = book)) +
```

```
  geom_col(show.legend = FALSE) +
  facet_wrap(~book, ncol = 2, scales = "free_x")
```




```
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)`
```

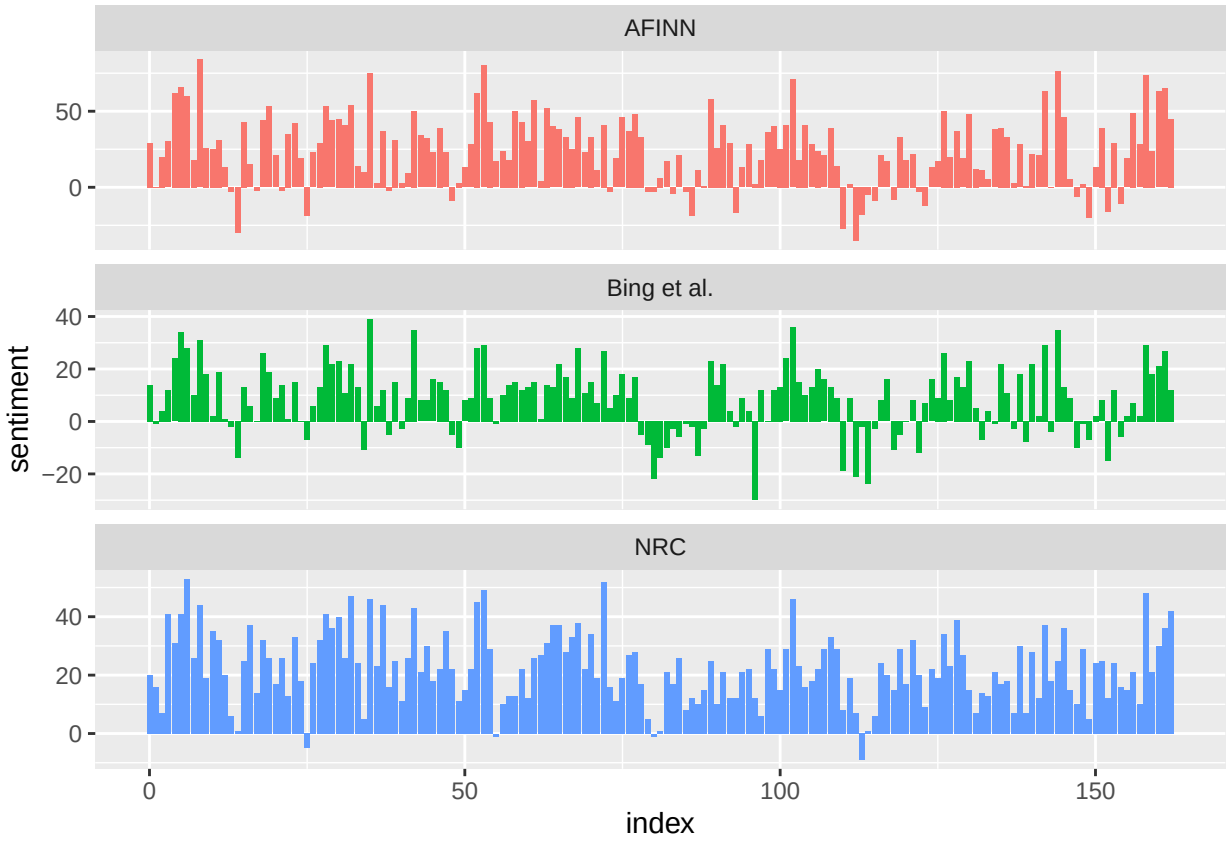
```
bing_and_nrc <- bind_rows(
  pride_prejudice %>%
    inner_join(get_sentiments("bing")) %>%
    mutate(method = "Bing et al."),
  pride_prejudice %>%
    inner_join(get_sentiments("nrc") %>%
                 filter(sentiment %in% c("positive",
                                         "negative"))
    ) %>%
    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.
```

```
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")
```




```
get_sentiments("nrc") %>%
  filter(sentiment %in% c("positive", "negative")) %>%
  count(sentiment)
```

```
## # A tibble: 2 x 2
##   sentiment     n
##   <chr>     <int>
## 1 negative   3316
```

```
## 2 positive   2308
```

```
get_sentiments("bing") %>%
  count(sentiment)
```

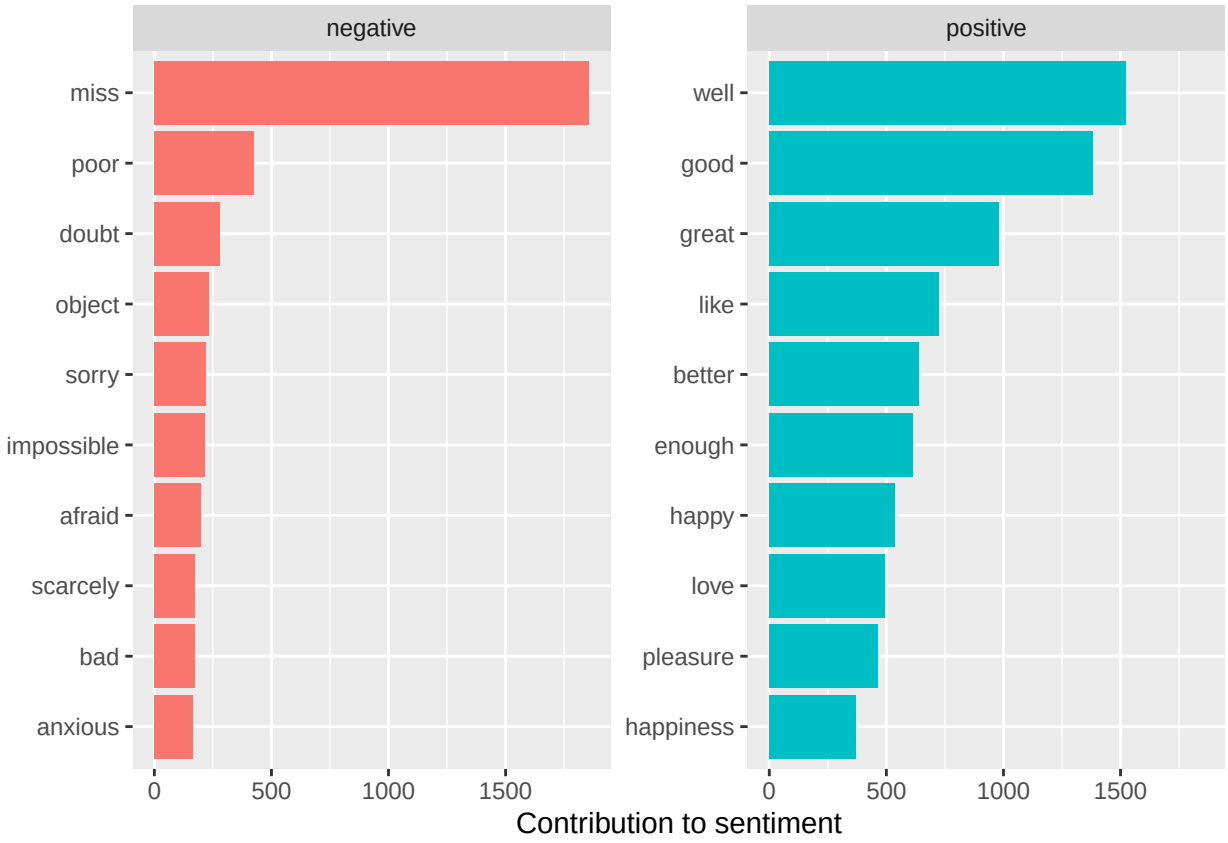
```
## # A tibble: 2 x 2
##   sentiment     n
##   <chr>     <int>
## 1 negative   4781
## 2 positive   2005
```

```
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)
```



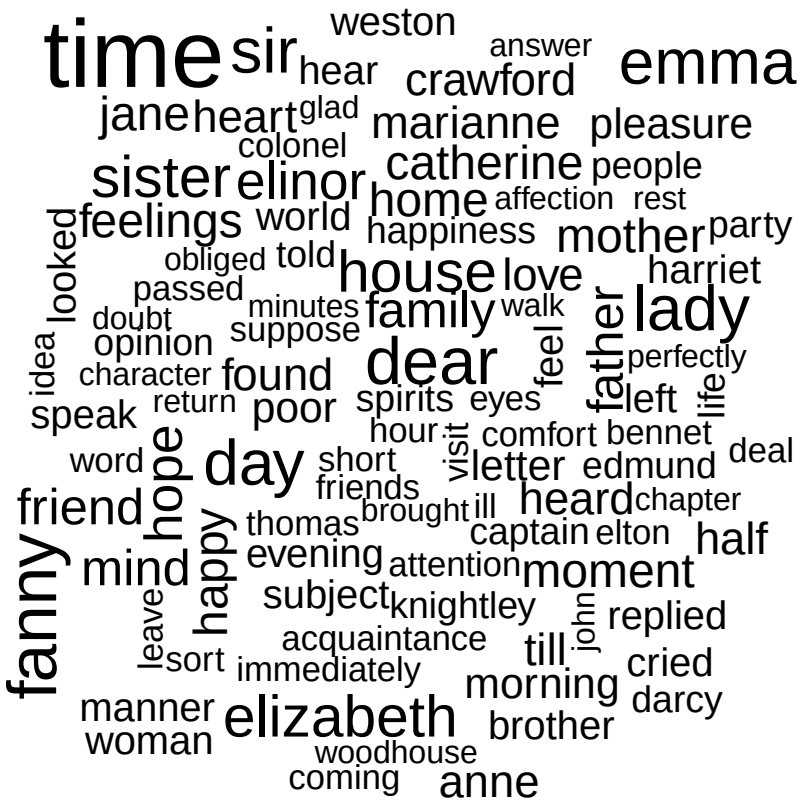

```
library(wordcloud)
```

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

```
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): miss could not be fit on page.
## It will not be plotted.
```




```
library(reshape2)
```

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

```
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.
```


```
p_and_p_sentences <- tibble(text = prideprejudice) %>%
  unnest_tokens(sentence, text, token = "sentences")

p_and_p_sentences$sentence[2]
```

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

```
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
```

We are going to use new corpus from package Gutenberg using author 'Luther, Martin'. We modify the data set so we can run sentiment analysis and dicover important words in these particular books

```
# load tthe book
library(gutenbergr)
# load the new corpus
Martin_books <- gutenberg_works(author == 'Luther, Martin')
head(Martin_books)
```

```
## # A tibble: 6 x 8
##   gutenberg_id title     author gutenberg_author_id language gutenberg_bookshelf
##          <int> <chr>     <chr>                <int> <chr>    <chr>
## 1          272 An Open ~ Luthe~                 155 en       Christianity
## 2          273 The Smal~ Luthe~                 155 en       Christianity
## 3          274 Disputat~ Luthe~                 155 en       Christianity/Harva~
## 4          418 A Treati~ Luthe~                 155 en       Christianity
## 5         1549 Commenta~ Luthe~                 155 en       Christianity
## 6         1670 Luther's~ Luthe~                 155 en       Christianity
## # i 2 more variables: rights <chr>, has_text <lgl>
```

```
tidy_martin <- Martin_books %>%
  gutenberg_download(meta_fields = 'title') %>%
  group_by(gutenberg_id) %>%
```

```
  mutate(linenumber = row_number()) %>%
  ungroup() %>%
  unnest_tokens(word, text)
```

```
## Determining mirror for Project Gutenberg from https://www.gutenberg.org/robot/harvest
```

```
## Using mirror http://aleph.gutenberg.org
```

```
martin_sent <- tidy_martin %>%

  inner_join(get_sentiments('bing'), by = 'word') %>%
  count(title, index = linenumber %/% 80, sentiment) %>%
  pivot_wider(names_from = sentiment, values_from = n, values_fill = 0) %>%
  mutate(sentiment = positive - negative)
```

```
## Warning in inner_join(., get_sentiments("bing"), by = "word"): Detected an unexpected many-to-many re
## i Row 145044 of `x` matches multiple rows in `y`.
## i Row 1185 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.
```

```
head(martin_sent)
```

```
## # A tibble: 6 x 5
##   title                     index negative positive sentiment
##   <chr>                     <dbl>    <int>    <int>     <int>
## 1 A Treatise on Good Works      0       18       29        11
## 2 A Treatise on Good Works      1       19       60        41
## 3 A Treatise on Good Works      2       21       92        71
## 4 A Treatise on Good Works      3       19       73        54
## 5 A Treatise on Good Works      4       15       61        46
## 6 A Treatise on Good Works      5       14       56        42
```

```
get_sentiments("nrc") %>%
  filter(sentiment %in% c("positive", "negative")) %>%
  count(sentiment)
```

```
## # A tibble: 2 x 2
##   sentiment     n
##   <chr>     <int>
## 1 negative   3316
## 2 positive   2308
```

```
get_sentiments("bing") %>%
  count(sentiment)
```

```
## # A tibble: 2 x 2
##   sentiment     n
##   <chr>     <int>
## 1 negative   4781
## 2 positive   2005
```

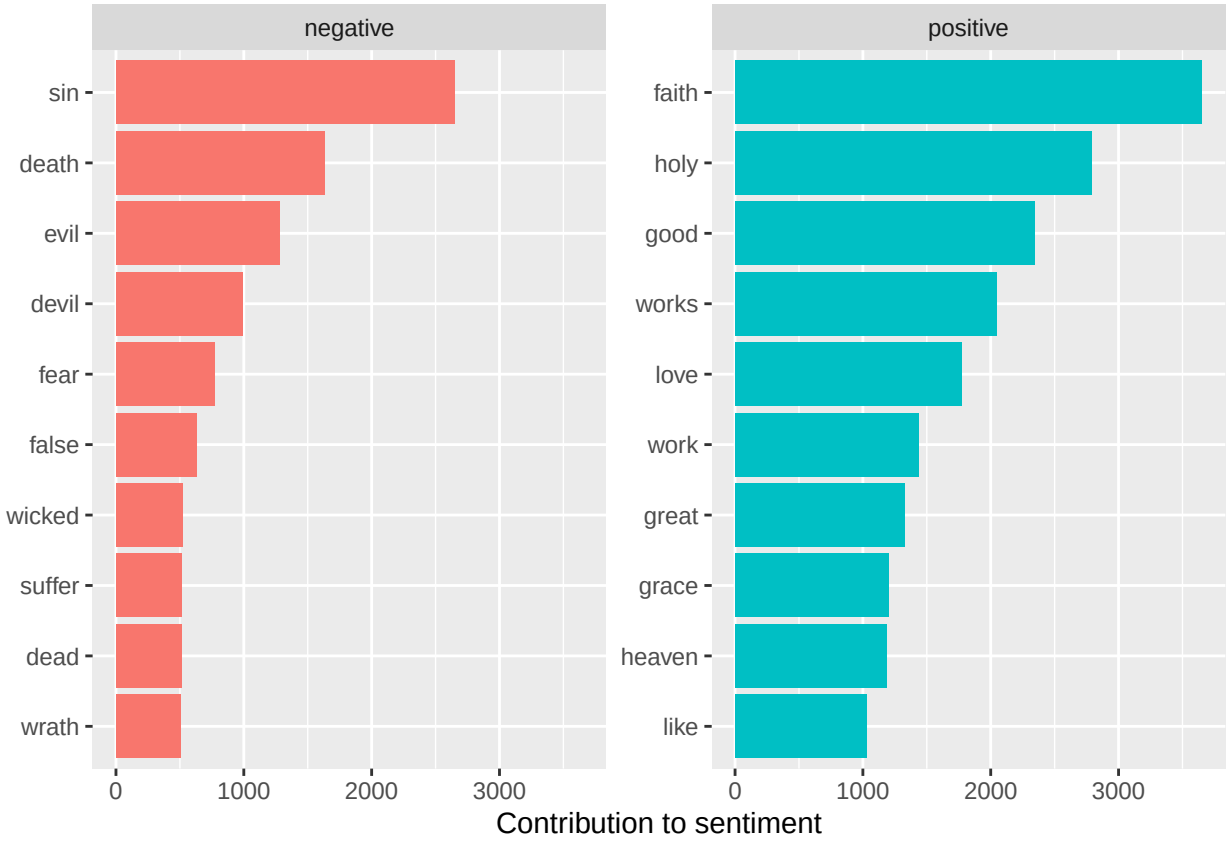
```
martin_word_counts <- tidy_martin %>% inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE) %>%
  ungroup()
```

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

```
## Warning in inner_join(., get_sentiments("bing")): Detected an unexpected many-to-many relationship be
```

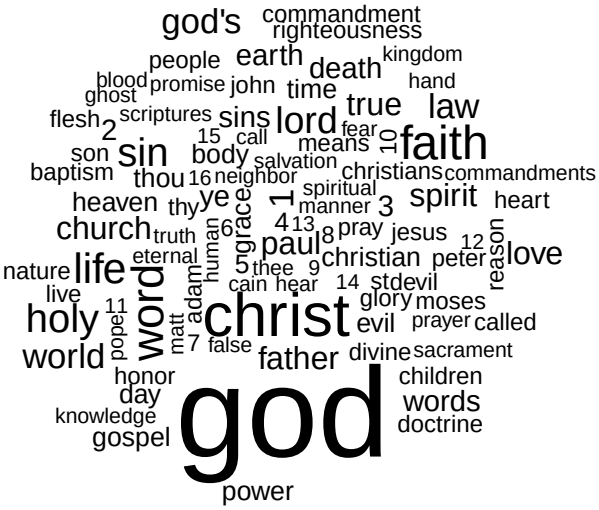
```
## i Row 145044 of `x` matches multiple rows in `y`.
## i Row 1185 of `y` matches multiple rows in `x`.
## i If a many-to-many relationship is expected, set `relationship =
##   "many-to-many"` to silence this warning.
```

```
martin_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)
```




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

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




```
library(reshape2)

tidy_martin %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE) %>%
  acast(word ~ sentiment, value.var = "n", fill = 0) %>%
  comparison.cloud(colors = c("blue", "green"),
                   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 145044 of `x` matches multiple rows in `y`.
## i Row 1185 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.
```


```
martin_words <- Martin_books %>%
  gutenberg_download(meta_fields = 'title') %>%
  group_by(gutenberg_id) %>%
  ungroup() %>%
  unnest_tokens(word, text) %>% rename(book = title) %>%
  count(book, word, sort = T)

total_words <- martin_words %>%
  group_by(book) %>%
  summarize(total = sum(n))
martin_words <- left_join(martin_words, total_words)
```

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

```
head(martin_words)
```

```
## # A tibble: 6 x 4
##   book                                                      word      n  total
##   <chr>                                                     <chr> <int>  <int>
## 1 Commentary on Genesis, Vol. 1: Luther on the Creation      the   12294 174847
## 2 Works of Martin Luther, with Introductions and Notes (Volu~ the   10682 170355
## 3 Commentary on Genesis, Vol. 2: Luther on Sin and the Flood  the   10054 129534
## 4 Epistle Sermons, Vol. 3: Trinity Sunday to Advent           the    8248 136454
## 5 Commentary on Genesis, Vol. 1: Luther on the Creation       of     7768 174847
## 6 Epistle Sermons, Vol. 2: Epiphany, Easter and Pentecost     the    7565 118888
```

```
book_tf_idf <- martin_words %>%
  bind_tf_idf(word, book, n)
book_tf_idf %>%
  select(-total) %>%
  arrange(desc(tf_idf))
```

```
## # A tibble: 78,805 x 6
##    book                                          word      n      tf   idf  tf_idf
##    <chr>                                         <chr> <int>   <dbl> <dbl>   <dbl>
##  1 Disputation of Doctor Martin Luther on the~ papa     19 0.00399 2.77  0.0111
##  2 Disputation of Doctor Martin Luther on the~ quod     23 0.00483 2.08  0.0100
##  3 Disputation of Doctor Martin Luther on the~ sunt     22 0.00462 2.08  0.00960
##  4 Disputation of Doctor Martin Luther on the~ veni~    15 0.00315 2.77  0.00873
##  5 Disputation of Doctor Martin Luther on the~ et       58 0.0122  0.693 0.00844
##  6 Disputation of Doctor Martin Luther on the~ pape     14 0.00294 2.77  0.00815
##  7 Disputation of Doctor Martin Luther on the~ est      31 0.00651 1.16  0.00757
##  8 Disputation of Doctor Martin Luther on the~ qui      17 0.00357 2.08  0.00742
##  9 Disputation of Doctor Martin Luther on the~ dei      11 0.00231 2.77  0.00640
## 10 Disputation of Doctor Martin Luther on the~ chri~    10 0.00210 2.77  0.00582
## # i 78,795 more rows
```

```
head(book_tf_idf)
```

```
## # A tibble: 6 x 7
##   book                                    word      n  total     tf   idf tf_idf
##   <chr>                                   <chr> <int>  <int>  <dbl> <dbl>  <dbl>
## 1 Commentary on Genesis, Vol. 1: Luther ~ the   12294 174847 0.0703     0      0
## 2 Works of Martin Luther, with Introduct~ the   10682 170355 0.0627     0      0
## 3 Commentary on Genesis, Vol. 2: Luther ~ the   10054 129534 0.0776     0      0
## 4 Epistle Sermons, Vol. 3: Trinity Sunda~ the    8248 136454 0.0604     0      0
## 5 Commentary on Genesis, Vol. 1: Luther ~ of     7768 174847 0.0444     0      0
## 6 Epistle Sermons, Vol. 2: Epiphany, Eas~ the    7565 118888 0.0636     0      0
```

```
library(forcats)
```

```
book_tf_idf %>%
  group_by(book) %>%
  slice_max(tf_idf, n = 15) %>%
  ungroup() %>%
  ggplot(aes(tf_idf, fct_reorder(word, tf_idf), fill = book)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~book, ncol = 2, scales = "free") +
  labs(x = "tf-idf", y = NULL)
```

These are the most common words in the novels written by Martin Luther. as we notice there are a lot of character names that are important for each corpus text with his novels.

References

Robinson, J. S. and D. (n.d.). 3 analyzing word and document frequency: Tf-IDF: Text mining with R. A Tidy Approach. https://www.tidytextmining.com/tfidf

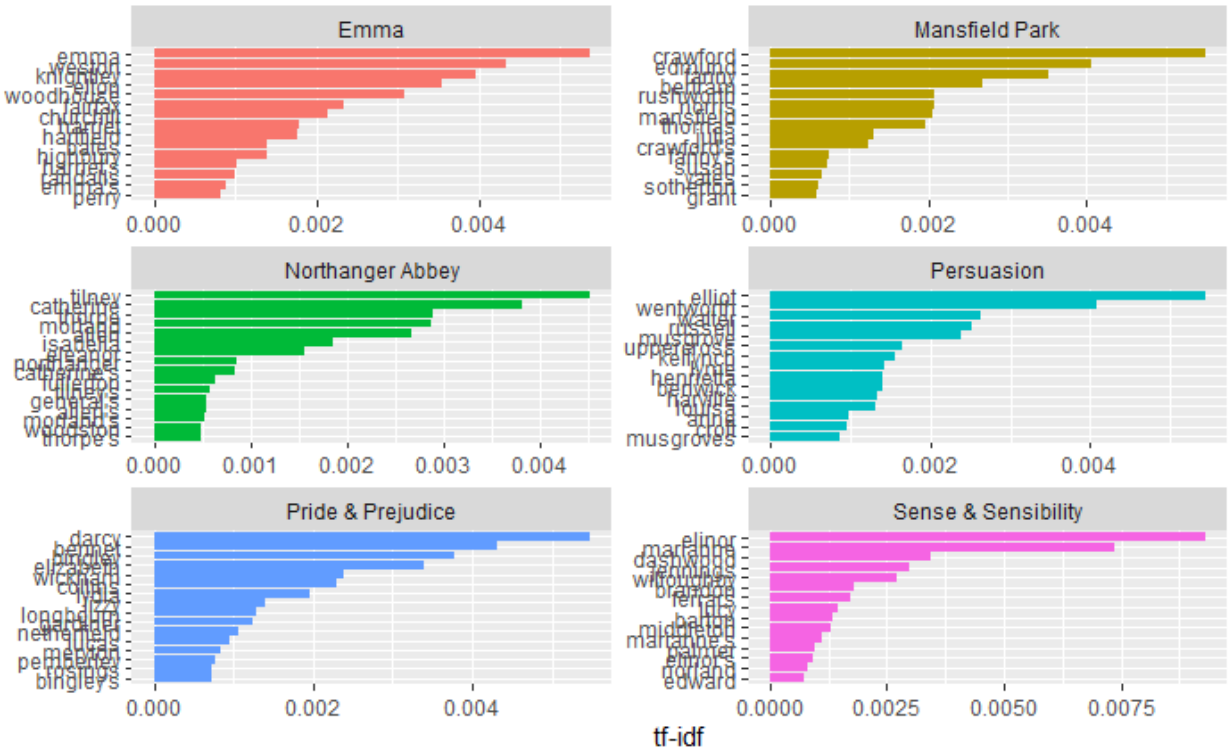


Figure 1: Most Important words