Text and sentiment analysis of My Ántonia (1918) by Willa Cather

Natural language processing conducted with R

Cherin-Juliette

10 september, 2022

Abstract

The novel My Ántonia, written by Willa Cather, follows the recollections of Jim Burden who leaves Virginia to live with his grandparents in rural Nebraska during the period of The Great Migration. Here he meets Ántonia Shimerda, a Bohemian immigrant who both embodies the shadow of his youth as well as the transformation of the prairie. Using the programming lanuage R, text from the novel is analyzed to detemine the lexical differences between the 5 books. In addition to text analysis, an analysis of sentiment is also conducted using the Bing, nrc and afinn lexicon.

Keywords: Ántonia, The Great Migration, R, Sentiment analysis

Contents

1 Making a custom ggplot2 theme

Within R graphs may be generated with the ggplot2 package which offers the user ways to make different type of graphs. This may be adapted and customized to one's liking, and in this instance I have chosen to transform a standard and boring graph into one which matching colors and aesthetics.

```
my_theme <- function() {
  theme_apa(legend.pos="none") +
    theme(panel.background=element_blank(),                # remove plot grids
          plot.background=element_rect(fill="#f3faff"),    # background color
          panel.border=element_blank(),                    # facet border
          strip.background=element_blank(),                # facet title background
          plot.margin = unit(c(.5, .5, .5, .5), "cm"))
}
```

2 Loading the required packages for the analysis

```
# Running this line of code will load the package if the user has previously installed it.
# If the package was not installed it will be downloaded from CRAN.

# devtools::install_github("cherjuliette/rcolorUtrecht")
if(!require(rcolorUtrecht)) install.packages("rcolorUtrecht")   # provides color palettes
if(!require(ggchicklet)) install.packages("ggchicklet")         # rounded bar plots
if(!require(gutenbergr)) install.packages("gutenbergr")         # the gutenbergr database
if(!require(tidyverse)) install.packages("tidyverse")
if(!require(tidytext)) install.packages("tidytext")             # making text data tidy
if(!require(cowplot)) install.packages("cowplot")               # combining multiple plots
if(!require(jtools)) install.packages("jtools")                 # regression analysis
```

3 Downloading the novel My Ántonia (1918) from the Gutenberg database

The gutenberg.org database is an open-source platform from which novels can be downloaded free of charge. Within R, the package gutenbergr allows the user to download and import novels from the gutenberg database into R without having to manually go to the website, download it and import it into R.

```
# check what written material by author Willa Cather is available on gutenberg.org
gutenberg_works(author=="Cather, Willa")
```

```
## # A tibble: 9 x 8
##   gutenberg_id title               author guten~1 langu~2 guten~3 rights has_t~4
##          <int> <chr>               <chr>    <int> <chr>   <chr>   <chr>  <lgl>
## 1           24 O Pioneers!         Cathe~      22 en      <NA>    Publi~ TRUE
## 2           44 The Song of the La~ Cathe~      22 en      Opera   Publi~ TRUE
## 3           94 Alexander's Bridge  Cathe~      22 en      <NA>    Publi~ TRUE
## 4          242 My Antonia          Cathe~      22 en      <NA>    Publi~ TRUE
## 5          346 The Troll Garden, ~ Cathe~      22 en      <NA>    Publi~ TRUE
## 6         2369 One of Ours         Cathe~      22 en      <NA>    Publi~ TRUE
## 7        13555 Youth and the Brig~ Cathe~      22 en      <NA>    Publi~ TRUE
## 8        19810 My Ántonia          Cathe~      22 en      <NA>    Publi~ TRUE
```

```
## 9        25586 A Collection of St~ Cathe~      22 en      <NA>    Publi~ TRUE
## # ... with abbreviated variable names 1: gutenberg_author_id, 2: language,
## #   3: gutenberg_bookshelf, 4: has_text
```

```
# download the novel into R
df <- gutenberg_download(242, mirror = "http://aleph.gutenberg.org")
df
```

```
## # A tibble: 8,275 x 2
##    gutenberg_id text
##           <int> <chr>
##  1          242 "MY ÁNTONIA"
##  2          242 ""
##  3          242 "By Willa Cather"
##  4          242 ""
##  5          242 ""
##  6          242 ""
##  7          242 ""
##  8          242 "TO CARRIE AND IRENE MINER"
##  9          242 ""
## 10          242 "In memory of affections old and true"
## # ... with 8,265 more rows
```

4 Prepare the data for analyis

Now that the novel My Ántonia is downloaded and imported into R, it is stored as an object named df (short for data frame). Before the text and sentiment analysis can be conducted the data must be prepared, which is called data cleaning and tidying. When data is cleaned and made tidy we remove information that is now necessary, such as the message from the author before the novel starts with which they dedicate the novel to a person, or the table of contents. This information adds very little value to an analysis and when removed gives the data a cleaner look. Afterwards, a column is added which displays which book the text belongs to. The novel is made up from 5 books. In each book Jim recalls his life on the prairie when he is at a different life stage (child, teenager, young-adult and adult). This will make looking for differences in lexical diversity and word count between the books easier.

```
df <- df %>%
  # remove the first 75 line
  slice(-(1:75)) %>%
  # select(-gutenberg_id) %>%
  # add a column with line number
  mutate(line_number=row_number(),
         # add a column with book name and number
         book_name=cumsum(str_detect(text, regex("^BOOK [\\divxlc]", ignore_case = TRUE))),
         book_number=cumsum(str_detect(text, regex("^BOOK [\\divxlc]", ignore_case = TRUE))))

# specify the book name
df$book_name[which(df$book_name=="0")]="Introduction"
df$book_name[which(df$book_name=="1")]="Book 1: The Shimerdas"
df$book_name[which(df$book_name=="2")]="Book 2: The Hired Girls"
df$book_name[which(df$book_name=="3")]="Book 3: Lena Lingard"
df$book_name[which(df$book_name=="4")]="Book 4: The Pioneer Woman's Story"
df$book_name[which(df$book_name=="5")]="Book 5: Cuzak's Boys"

# view(df)
```

5 Text analysis

The first step of text analysis using R is making the text tidy. This is done by splitting up every sentence into separate words in a process named tokenization. Now each word receives its own token which may later be counted.

5.1 Find the most common punctuations used per book

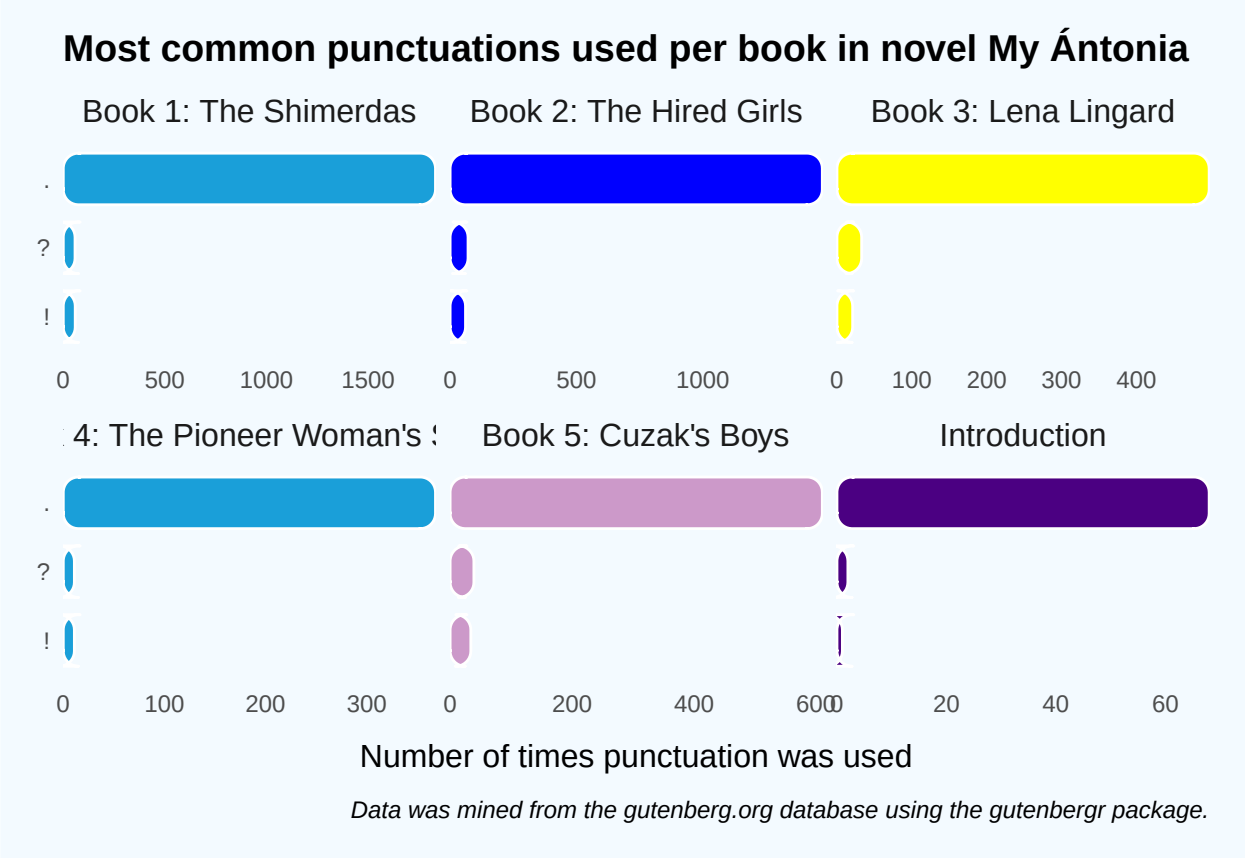
I thought it would be interesting to see which punctuation were most commonly used. It doesn't necessarily say much about the novel and I suspected that Jim, the narrator, in his recollections of his youth would not often ask himself questions. For this reason I thought the most commonly used punctuation would be a full stop, which as it turns out is true.

```
# filter punctuations from text
punctuation <- df %>%
  # break up the text of the novel into separate words (=token)
  unnest_tokens(token, text, strip_punct = F) %>%
  # count the number of punctuations per book
  count(book_name, token, sort = T) %>%
  filter(token %in% c("!", "?", "."))
punctuation
```

```
## # A tibble: 18 x 3
##    book_name                          token     n
##    <chr>                              <chr> <int>
##  1 Book 1: The Shimerdas              .      1832
##  2 Book 2: The Hired Girls            .      1472
##  3 Book 5: Cuzak's Boys               .       610
##  4 Book 3: Lena Lingard               .       497
##  5 Book 4: The Pioneer Woman's Story  .       368
##  6 Book 2: The Hired Girls            ?        71
##  7 Introduction                       .        68
##  8 Book 2: The Hired Girls            !        61
##  9 Book 1: The Shimerdas              !        59
## 10 Book 1: The Shimerdas              ?        58
## 11 Book 5: Cuzak's Boys               ?        39
## 12 Book 5: Cuzak's Boys               !        34
## 13 Book 3: Lena Lingard               ?        33
## 14 Book 3: Lena Lingard               !        21
## 15 Book 4: The Pioneer Woman's Story  !        11
## 16 Book 4: The Pioneer Woman's Story  ?        11
## 17 Introduction                       ?         2
## 18 Introduction                       !         1
```

```
# visualize the results
punctuation %>%
  # put the most used punctuation on top
  mutate(token = reorder(token, n)) %>%
  ggplot(mapping=aes(x=token, y=n, fill=book_name)) +
  geom_chicklet(width = 0.75,
                radius=grid::unit(2,"mm")) +
  facet_wrap(~book_name, scales = "free_x") +
  coord_flip() +
  scale_y_continuous(expand = c(0,0)) +
```

```
  my_theme() +
  theme(axis.ticks.y=element_blank(),
        axis.ticks.x=element_blank(),
        axis.title.y=element_blank(),
        axis.title.x=element_text(margin = margin(10,0,0,0)),
        plot.caption=element_text(face="italic", margin = margin(10,0,0,0))) +
  ggtitle("Most common punctuations used per book in novel My Ántonia") +
  labs(caption="Data was mined from the gutenberg.org database using the gutenbergr package.",
       y="Number of times punctuation was used") +
  scale_fill_rcolorUtrecht(palette = "hu")
```



5.2 Counting the number of sentences per book

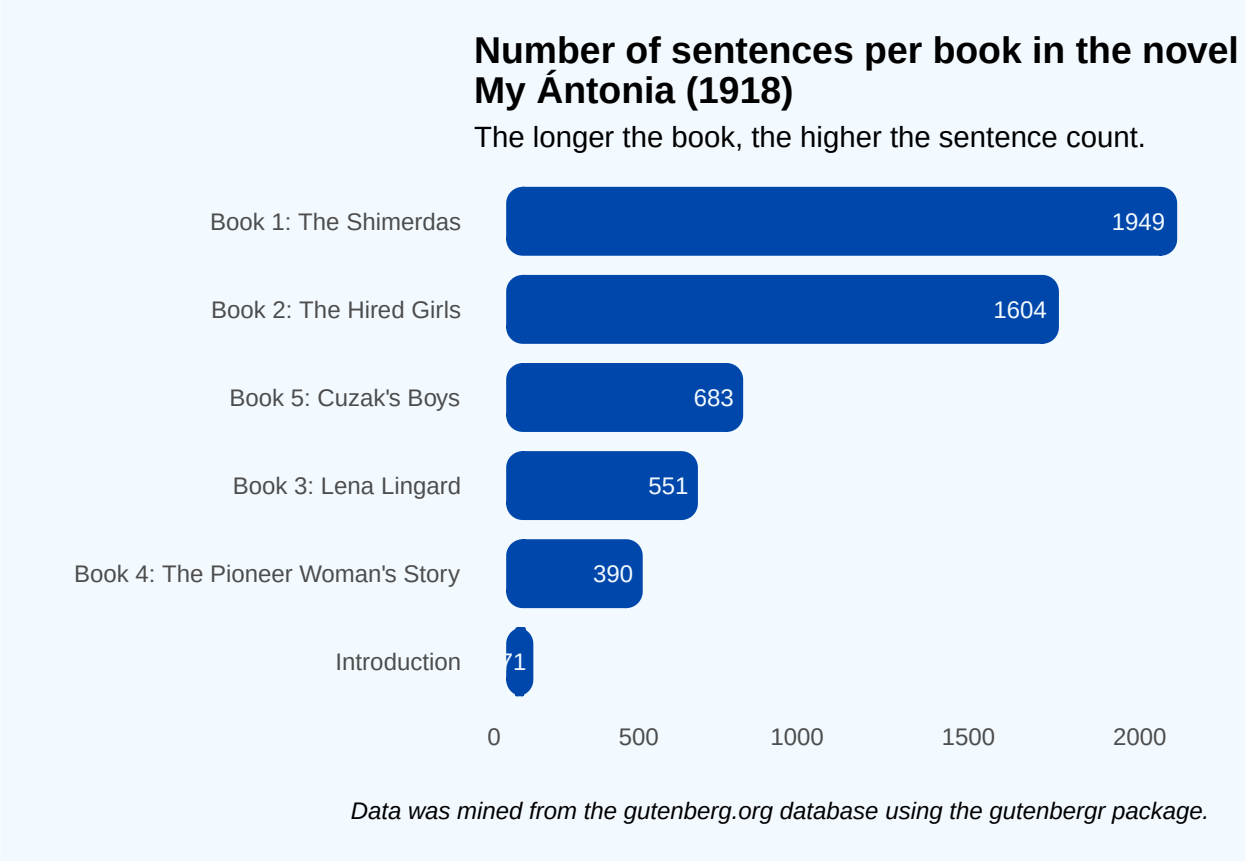
Now that the most used punctuation types are established the number of sentences can also be counted. A sentence ends when the punctuation is introduced.

```
# calculate the number of sentences
total_sentences <- punctuation %>%
  group_by(book_name) %>%
  summarize(total_sentences = sum(n))
total_sentences
```

```
## # A tibble: 6 x 2
##   book_name                      total_sentences
##   <chr>                                    <int>
## 1 Book 1: The Shimerdas                     1949
```

```
## 2 Book 2: The Hired Girls                      1604
## 3 Book 3: Lena Lingard                          551
## 4 Book 4: The Pioneer Woman's Story             390
## 5 Book 5: Cuzak's Boys                          683
## 6 Introduction                                   71
```

```
# visualize the results
total_sentences %>%
  ggplot(mapping=aes(x=reorder(book_name, total_sentences),
                     y=total_sentences,
                     fill = book_name)) +
  geom_chicklet(width=0.75,
                radius=grid::unit(2,"mm"),
                fill="#0047ab", color="#0047ab") +
  my_theme() +
  coord_flip() +
  geom_text(aes(label=total_sentences), vjust=0.5, hjust=1.2, color="#f3faff", size=3) +
  theme(axis.ticks.y=element_blank(),
        axis.ticks.x=element_blank(),
        plot.caption=element_text(face="italic"),
        axis.text.x = element_text(hjust=1.5)) +
  ggtitle("Number of sentences per book in the novel\nMy Ántonia (1918)") +
  labs(x="", y="",
       subtitle="The longer the book, the higher the sentence count.",
       caption = "Data was mined from the gutenberg.org database using the gutenbergr package.")
```



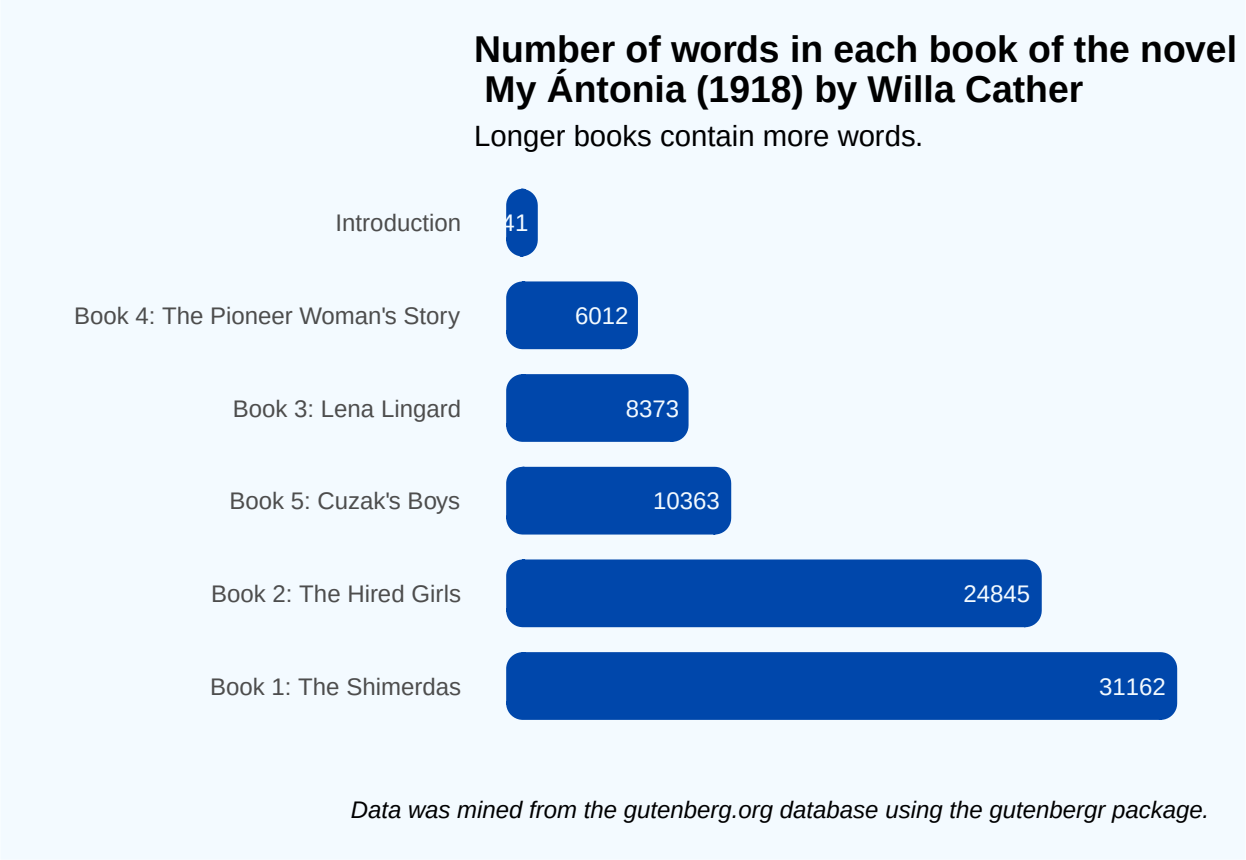
```
  # scale_fill_rcolorUtrecht(palette = "hu")
```

5.3 Counting the number of words per book

```
# count the number of words per book
total_words<-df %>%
  group_by(book_name) %>%
  unnest_tokens(output=word, input=text, token="words") %>%
  summarize(total_word_count=n())
total_words
```

```
## # A tibble: 6 x 2
##   book_name                          total_word_count
##   <chr>                                         <int>
## 1 Book 1: The Shimerdas                         31162
## 2 Book 2: The Hired Girls                       24845
## 3 Book 3: Lena Lingard                           8373
## 4 Book 4: The Pioneer Woman's Story              6012
## 5 Book 5: Cuzak's Boys                          10363
## 6 Introduction                                   1341
```

```
# visualize the results
total_words %>%
  ggplot(mapping=aes(x=reorder(book_name, -total_word_count),
                     y=total_word_count)) +
  geom_chicklet(width=0.7,
                radius=grid::unit(2,"mm"),
                fill="#0047ab", color="#0047ab") +
  coord_flip() +
  my_theme() +
  geom_text(aes(label=total_word_count),
            hjust=1.15,
            color="#f3faff",
            size=3) +
  theme(axis.ticks.x=element_blank(),
        axis.ticks.y=element_blank(),
        plot.caption=element_text(face="italic"),
        axis.text.x=element_blank()) +
  labs(subtitle = "Longer books contain more words.",
       caption = "Data was mined from the gutenberg.org database using the gutenbergr package.",
       x="", y="") +
  ggtitle("Number of words in each book of the novel \n My Ántonia (1918) by Willa Cather")
```



In Book 1: The Shimerdas Jim tells of his arrival in Nebraska when he goes to live with his grandparents on the prairie and meets the Shimerda family. Ántonia Shimerda, a 15 year old Bohemian immigrant, quickly befriends Jim and that initial friendship ensures the making Ántonia into an emblem of the prairie and his youth. For this reason I believe this chapter is the longest in the novel, which subsequently also means that this chapter has the highest number of sentences and words.

5.4 Comparing lexical diversity per book

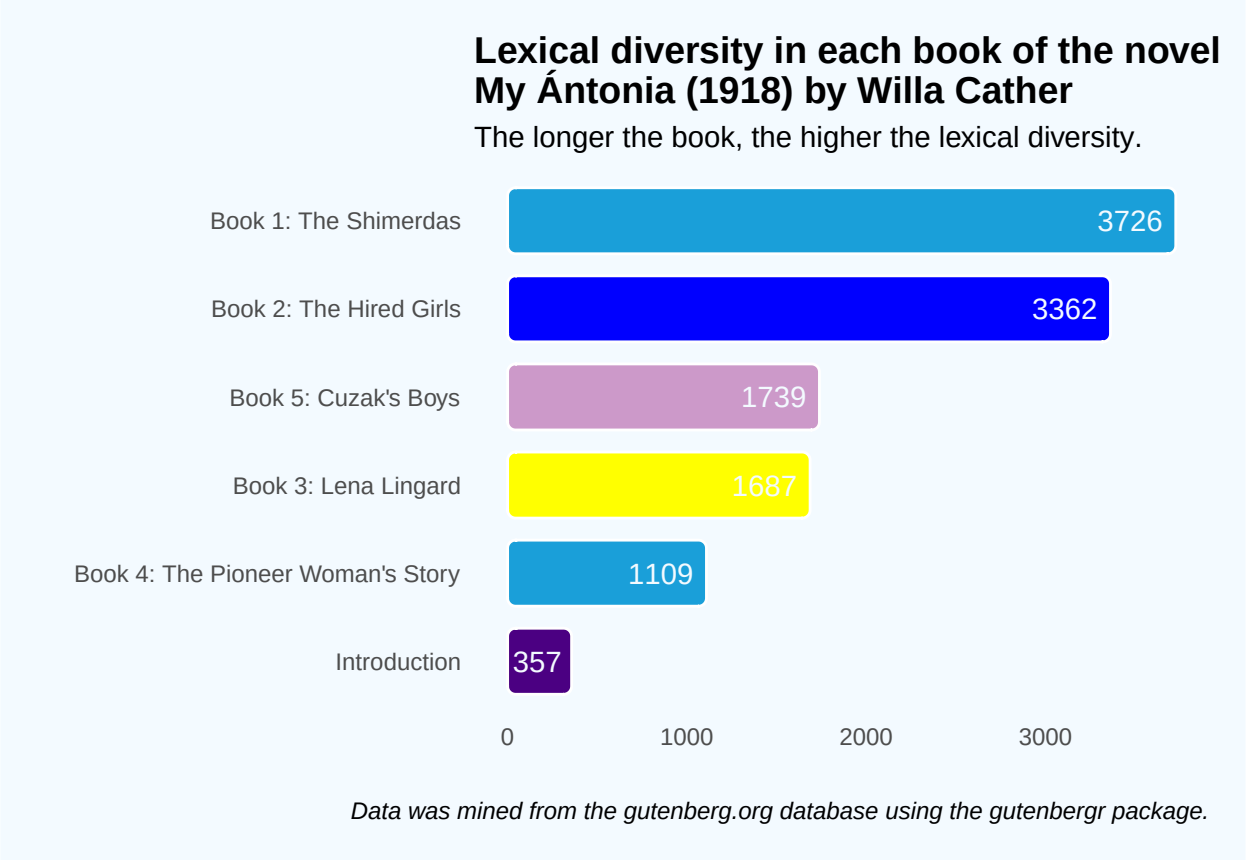
Lexical diversity is not how many words are used but how many unique words are used. With the n_distinct function I calculate the number of unique words are used per book in the novel.

```
# compare the number of unique words (lexical diversity) per book
lexical_diversity<-df %>%
  group_by(book_name) %>%
  unnest_tokens(input=text, output=word) %>%
  anti_join(stop_words, by="word") %>%
  count(book_name, word, sort=T) %>%
  summarize(lex_diversity=n_distinct(word))
lexical_diversity
```

```
## # A tibble: 6 x 2
##   book_name                          lex_diversity
##   <chr>                                      <int>
## 1 Book 1: The Shimerdas                       3726
## 2 Book 2: The Hired Girls                     3362
## 3 Book 3: Lena Lingard                        1687
## 4 Book 4: The Pioneer Woman's Story           1109
```

```
## 5 Book 5: Cuzak's Boys                          1739
## 6 Introduction                                   357
```

```r
# visualize the results
lexical_diversity %>%
  ggplot(mapping=aes(x=reorder(book_name, lex_diversity),
                     y = lex_diversity,
                     fill = book_name)) +
  geom_chicklet(width = 0.75) +
  scale_fill_rcolorUtrecht(palette = "hu") +
  coord_flip() +
  my_theme() +
  labs(x="", y="",
       subtitle="The longer the book, the higher the lexical diversity.",
       caption = "Data was mined from the gutenberg.org database using the gutenbergr package.") +
  geom_text(aes(label=lex_diversity), hjust=1.2, color="#f3faff", size=4) +
  ggtitle("Lexical diversity in each book of the novel\nMy Ántonia (1918) by Willa Cather") +
  theme(axis.ticks.x=element_blank(),
        axis.ticks.y=element_blank(),
        # axis.text.y=element_blank(),
        plot.caption=element_text(face="italic"))
```

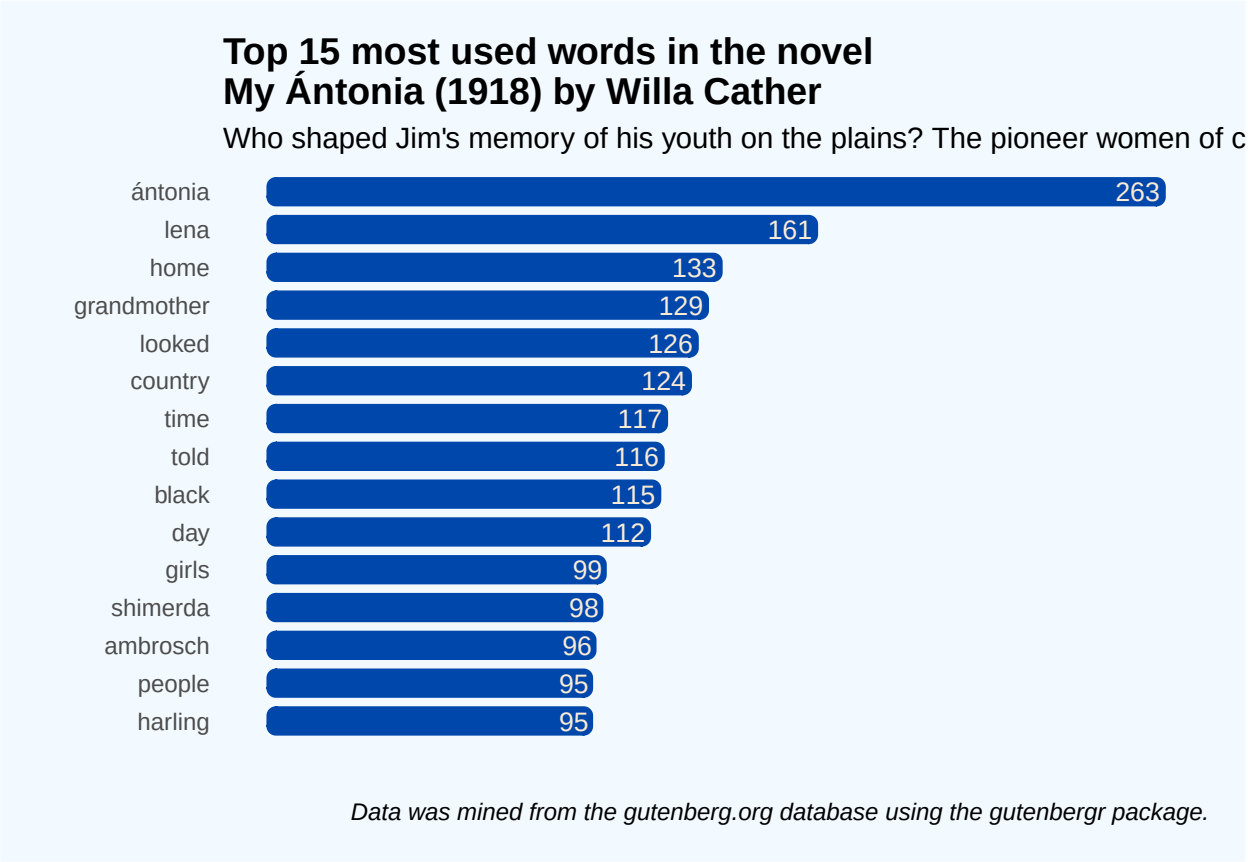


5.5 Finding the top 15 most used words used in the novel

Which words were most used in the novel? Why is that important? The reason why I was interested in this is when the words most often used is determined, I can also get a clearer picture of who or what Jim has made into reference points when thinking of his past.

First I have again split up all text in the novel into separate words and given them an ID (=token). Then I remove stop words from the text. Stop words such as she'd, there, is and super are often used in text and speech but tell very little about the core message of what we are trying to say. So to prevent these words from cluttering up the code, I remove them with the anti_join() function and count each word. The 15 words most often used will then be put in a graph.

```
df %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words) %>%
  filter(!word=="house",
         !word=="head",
         !word=="don't") %>%
  count(word, sort = T) %>%
  head(n = 15) %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(word, n)) +
  geom_chicklet(width = 0.7,
                fill="#0047ab", color="#0047ab") +
  coord_flip() +
  geom_text(aes(label=n),
            hjust=1.1,
            color="antiquewhite",
            size=3.5) +
  my_theme() +
  labs(x="", y="",
       subtitle="Who shaped Jim's memory of his youth on the plains? The pioneer women of course.",
       caption = "Data was mined from the gutenberg.org database using the gutenbergr package.") +
    theme(axis.ticks.x=element_blank(),
          axis.ticks.y=element_blank(),
          axis.text.x=element_blank(),
          plot.caption=element_text(face="italic")) +
  ggtitle("Top 15 most used words in the novel\nMy Ántonia (1918) by Willa Cather")
```



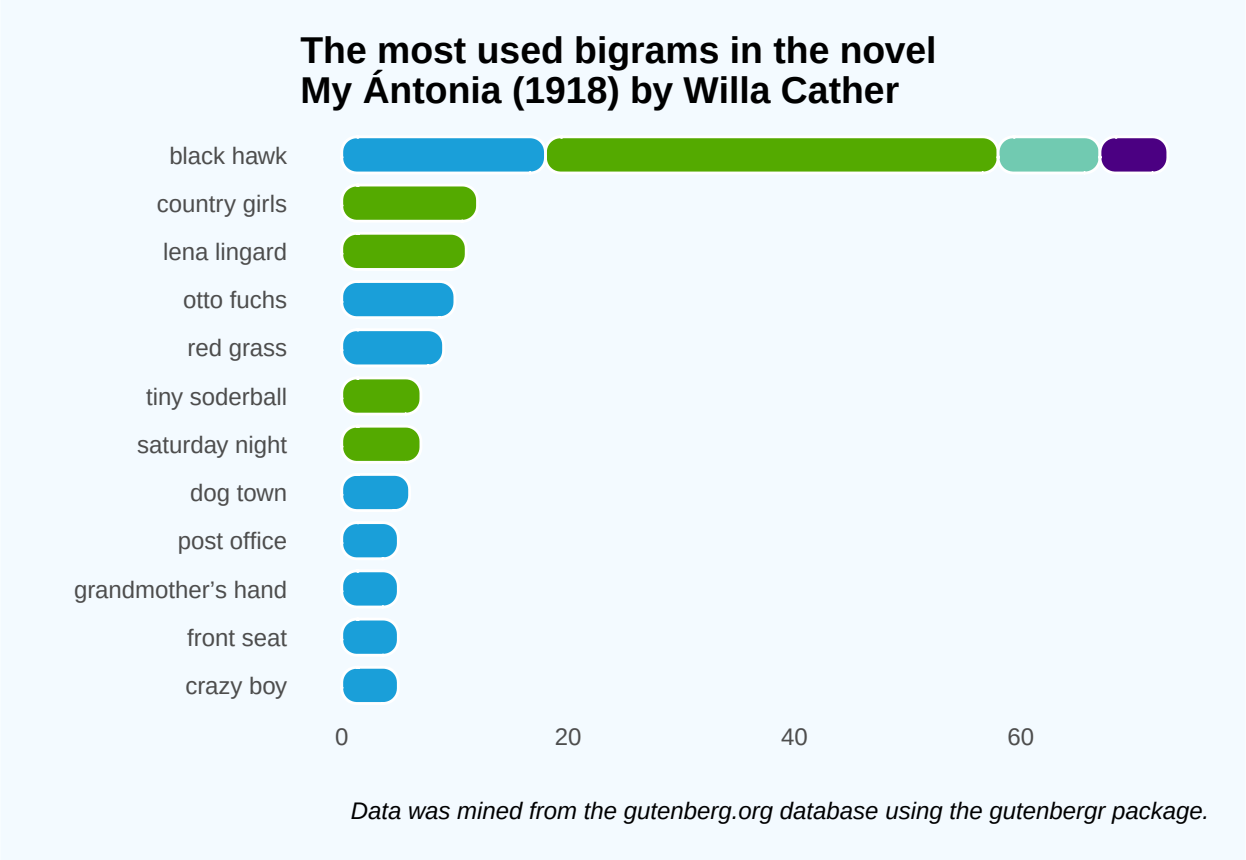
The word and name Ántonia is the number one word used by Jim in the novel. This was suspected considering how great of a symbol she is in the novel. While Jim's warm feelings towards Lena was greatly expressed in the novel it did surprise me that her name was used as frequently as it is.

5.6 Word correlations by finding the most used bigrams in the novel

Similarly to the function I used above to calculate the top 15 of most frequently used words, I can do the same thing but calculate the most common bigrams. Bigrams can convey a different message from singular words, namely how words are correlated to one another or which locations are commonly referred to.

```
bigrams<-df %>%
  group_by(book_name) %>%
  unnest_tokens(bigram, text, token="ngrams", n=2) %>%
  count(bigram, sort=T)

bigrams_united<-bigrams %>%
  separate(bigram, into = c("word1", "word2"), sep = " ") %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word) %>%
  unite(bigram, c(word1, word2), sep = " ")

bigrams_united %>%
  group_by(book_name) %>%
  head(15) %>%
  ggplot(aes(reorder(bigram, n), n, fill=book_name)) +
  geom_chicklet(width=0.75,
                radius=unit(2,"mm")) +
  coord_flip() +
```

```
my_theme() + scale_fill_rcolorUtrecht(palette="hu") +
labs(x="", y="",
     caption="Data was mined from the gutenberg.org database using the gutenbergr package.") +
ggtitle("The most used bigrams in the novel\nMy Ántonia (1918) by Willa Cather") +
theme(axis.ticks.x=element_blank(),
      axis.ticks.y=element_blank(),
      plot.caption=element_text(face="italic"))
```



After running the code it seems that Black Hawk, the location were the Lingard family and the Cutter family resides as well as were many of the girls sought employement, is most commonly used in the novel. This location has great significance in the novel because it was were Jim met many of the girls and where the girls started working to support their families. Perhaps because of this reason the second most used bigram is country girls.

6 Sentiment analysis

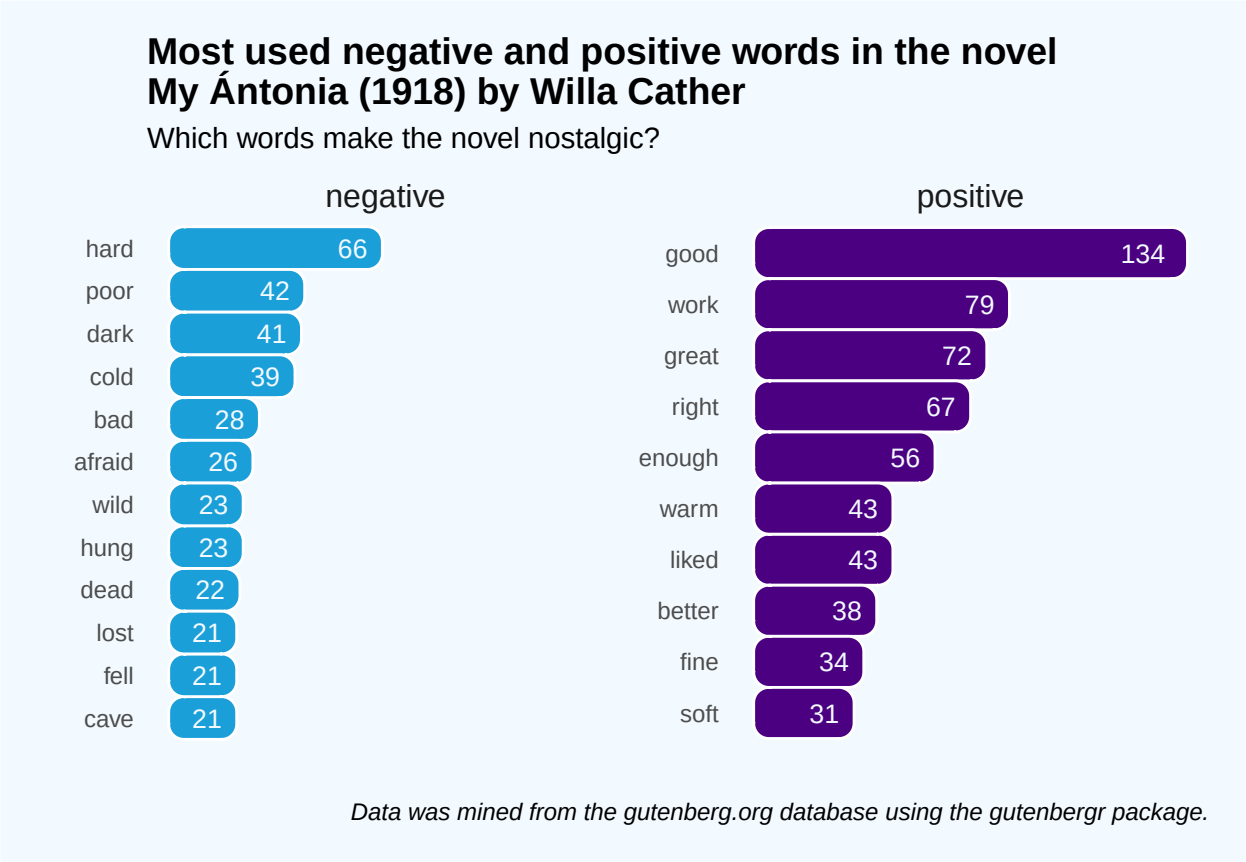
Now that the text is dissected I want to perform a sentiment analysis. Words on their own only possess the meaning we place upon them, however in recent years programmers specializing in Natural Language Processing have made several collections of words and evaluated the emotion that is paired with them. This is primarily done to evaluate the emotional index of Twitter posts (see articles A Sentiment Analysis of President Trump's Inaugural Address and Text analysis of Trump's tweets confirms he writes only the Android half). However the same method can be applied to any text, even novels.

The three collections of words and their emotional index are + AFINN by Finn Årup Nielsen + bing et al. by Bing Liu and collaborators + nrc by Saif Mohammad and Peter Turney and + loughran by Loughran and McDonald which is primarily used for economic and financial terms.

Is this definitive? Not exactly. The reason for this is because we must also keep in thought that the connotation of certain words have changed over time. An example of this is the word *miss* which according to all lexicons mentioned above is a word with a negative connotation. However this not necessary negative. For this reason I first determine which words with a negative and positive connotation are most frequently seen in the novel.

6.1 Finding the most used words with a negative and positive connotation

```
neg_pos_plot<-df %>%
  unnest_tokens(word, text) %>%
  anti_join(get_stopwords(), by="word") %>%
  filter(!word=="like",
         !word=="burden",
         !word=="well") %>%
  inner_join(get_sentiments("bing"), by="word") %>%
  count(word, sentiment, sort = T) %>%
  # ungroup() %>%
  group_by(sentiment) %>%
  top_n(10) %>%
  ungroup() %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(word, n, fill = sentiment)) +
  geom_chicklet(width = 1,
                radius=unit(2,"mm")) +
  facet_wrap(~sentiment, scales = "free_y") +
  coord_flip() +
  my_theme() +
  geom_text(aes(label=n), hjust=1.5, color="#f3faff", size=3.5) +
  labs(x="", y="",
       subtitle="Which words make the novel nostalgic?",
       caption = "Data was mined from the gutenberg.org database using the gutenbergr package.") +
    theme(axis.ticks.x=element_blank(),
          axis.ticks.y=element_blank(),
          axis.text.x=element_blank(),
          plot.caption=element_text(face="italic")) +
  ggtitle("Most used negative and positive words in the novel\nMy Ántonia (1918) by Willa Cather")

neg_pos_plot + scale_fill_rcolorUtrecht(palette = "hu")
```

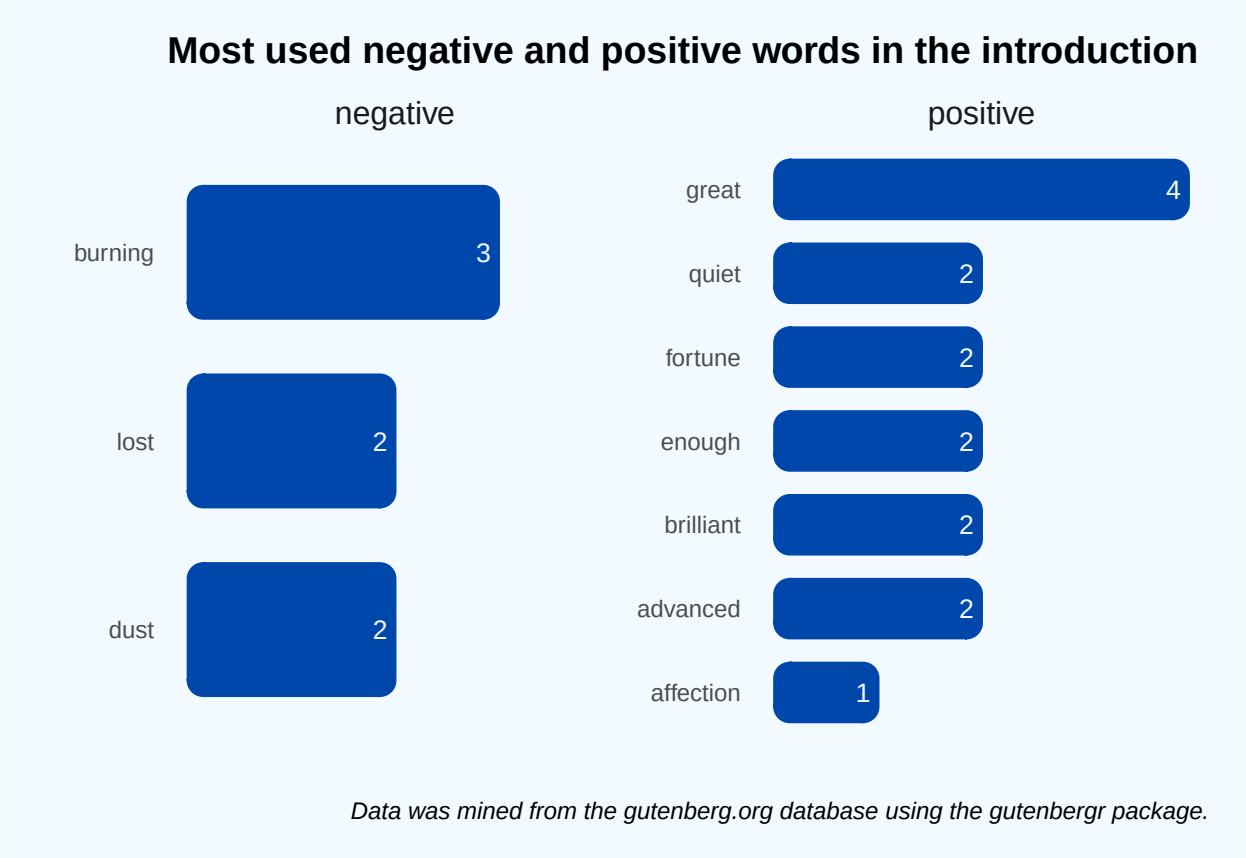


After running the code it becomes easier to understand which word distinctions make the novel more negative or positive. The most common positive words seen in the novel besides good are work and warm which pertain to a certain sense of security and comfort. In contrast the word poor is labelled as negative because it indicates a sense of instability both in life and financially.

This clarified much of the details of the novel for me, however I also wonder what I would find if I zoomed in and determined the most common negative and positive words per book in the novel.

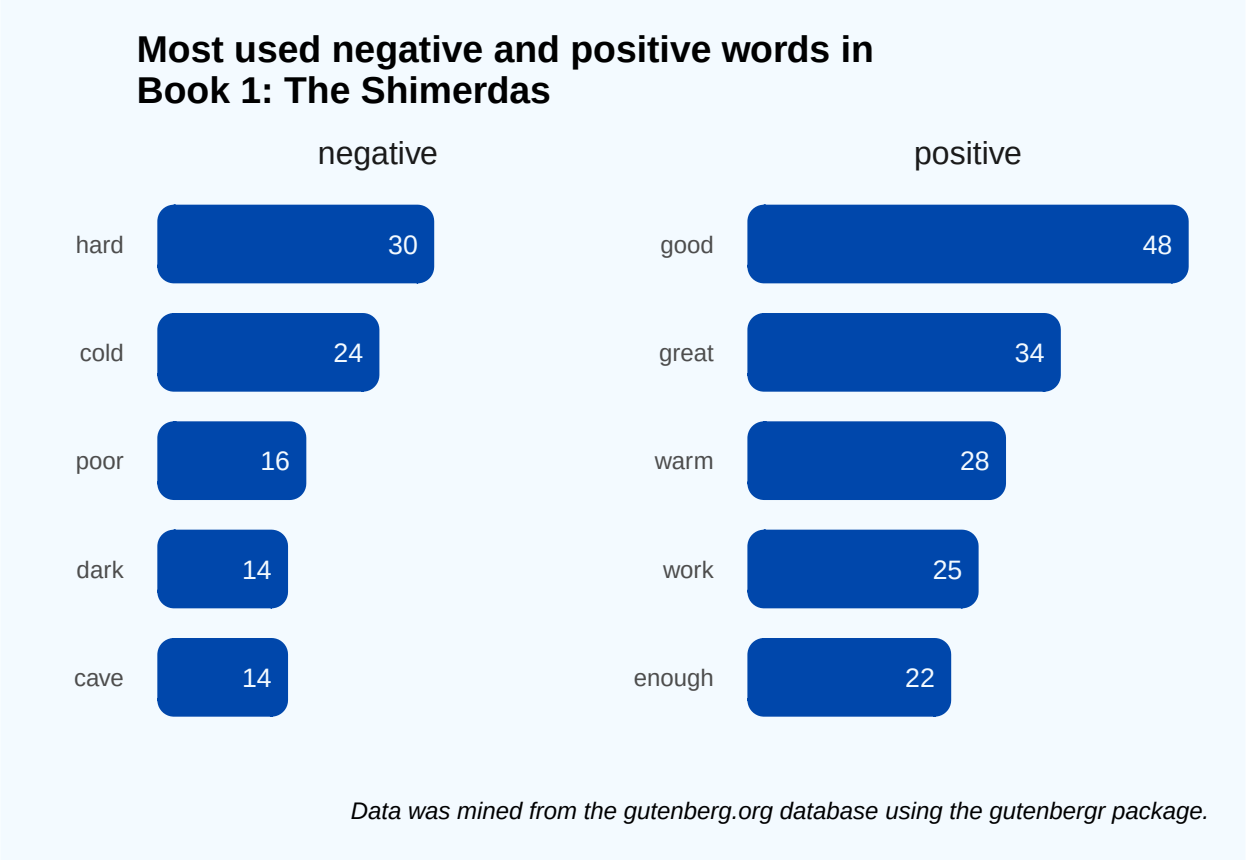
```
neg_pos_0<-df %>%
  filter(book_name=="Introduction") %>%
  unnest_tokens(word, text) %>%
  anti_join(get_stopwords()) %>%
  filter(!word=="like",
         !word=="burden",
         !word=="well") %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = T) %>%
  group_by(sentiment) %>%
  head(10) %>%
  # top_n(4) %>%
  ungroup() %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(word, n, fill = sentiment)) +
  geom_chicklet(width = 0.7,
                radius=unit(2,"mm"),
                fill="#0047ab", color="#0047ab") +
  facet_wrap(~sentiment, scales = "free_y") +
  coord_flip() +
  my_theme() +
```

```
  geom_text(aes(label=n), hjust=1.5, color="#f3faff", size=3.5) +
  labs(x="", y="",
       caption = "Data was mined from the gutenberg.org database using the gutenbergr package.") +
    theme(axis.ticks.x=element_blank(),
          axis.ticks.y=element_blank(),
          axis.text.x=element_blank(),
          plot.caption=element_text(face="italic")) +
  ggtitle("Most used negative and positive words in the introduction")
neg_pos_0
```



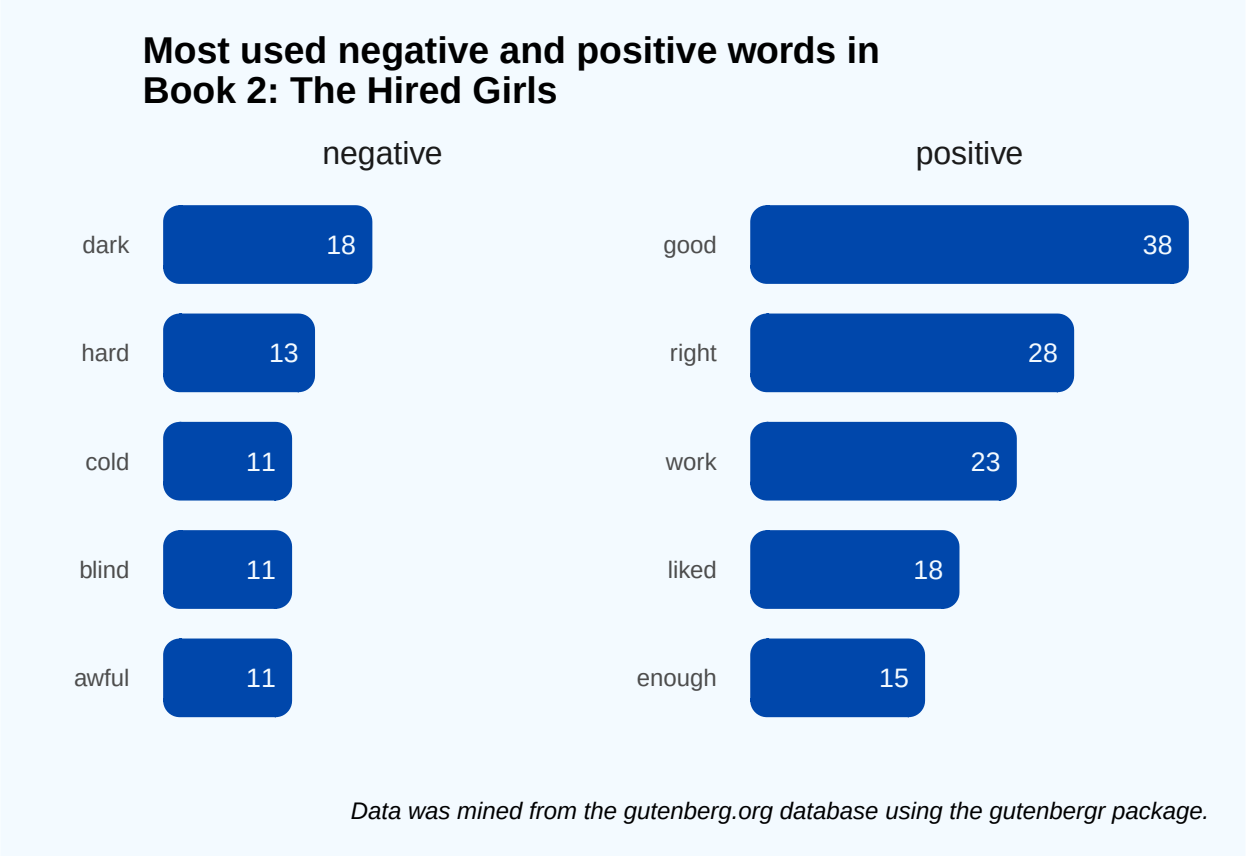

```
neg_pos_1<-df %>%
  filter(book_name=="Book 1: The Shimerdas") %>%
  unnest_tokens(word, text) %>%
  anti_join(get_stopwords()) %>%
  filter(!word=="like",
         !word=="burden",
         !word=="well") %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = T) %>%
  # ungroup() %>%
  group_by(sentiment) %>%
  # head(10) %>%
  top_n(5) %>%
  ungroup() %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(word, n, fill = sentiment)) +
  geom_chicklet(width = 0.7,
                radius=unit(2,"mm"),
```

```
                fill="#0047ab", color="#0047ab") +
  facet_wrap(~sentiment, scales = "free_y") +
  coord_flip() +
  my_theme() +
  geom_text(aes(label=n), hjust=1.5, color="#f3faff", size=3.5) +
  labs(x="", y="",
       caption = "Data was mined from the gutenberg.org database using the gutenbergr package.") +
    theme(axis.ticks.x=element_blank(),
          axis.ticks.y=element_blank(),
          axis.text.x=element_blank(),
          plot.caption=element_text(face="italic")) +
  ggtitle("Most used negative and positive words in\nBook 1: The Shimerdas")
neg_pos_1
```



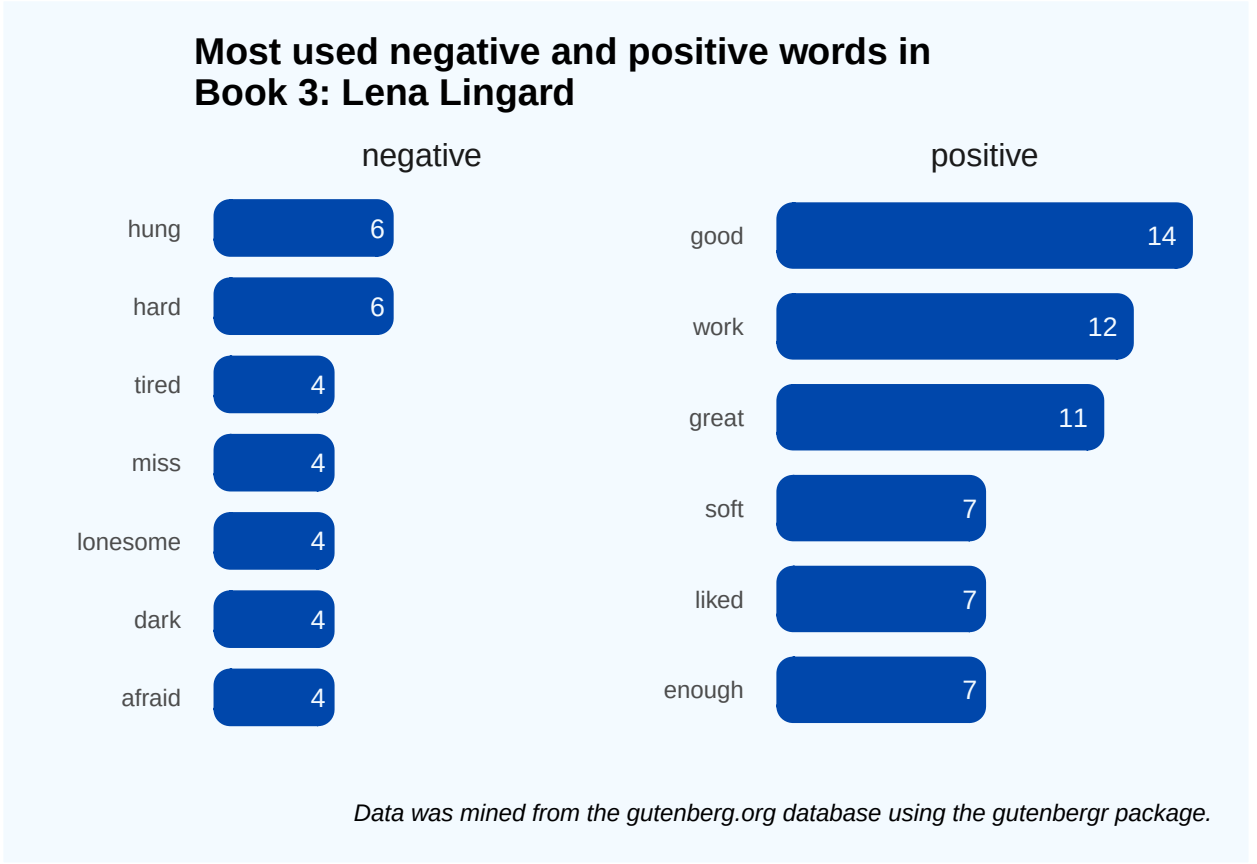

```
neg_pos_2<-df %>%
  filter(book_name=="Book 2: The Hired Girls") %>%
  unnest_tokens(word, text) %>%
  anti_join(get_stopwords()) %>%
  filter(!word=="like",
         !word=="burden",
         !word=="well") %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = T) %>%
  # ungroup() %>%
  group_by(sentiment) %>%
  # head(10) %>%
  top_n(5) %>%
  ungroup() %>%
```

```
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(word, n, fill = sentiment)) +
  geom_chicklet(width = 0.7,
                radius=unit(2,"mm"),
                fill="#0047ab", color="#0047ab") +
  facet_wrap(~sentiment, scales = "free_y") +
  coord_flip() +
  my_theme() +
  geom_text(aes(label=n), hjust=1.5, color="#f3faff", size=3.5) +
  labs(x="", y="",
       caption = "Data was mined from the gutenberg.org database using the gutenbergr package.") +
    theme(axis.ticks.x=element_blank(),
          axis.ticks.y=element_blank(),
          axis.text.x=element_blank(),
          plot.caption=element_text(face="italic")) +
  ggtitle("Most used negative and positive words in\nBook 2: The Hired Girls")
neg_pos_2
```



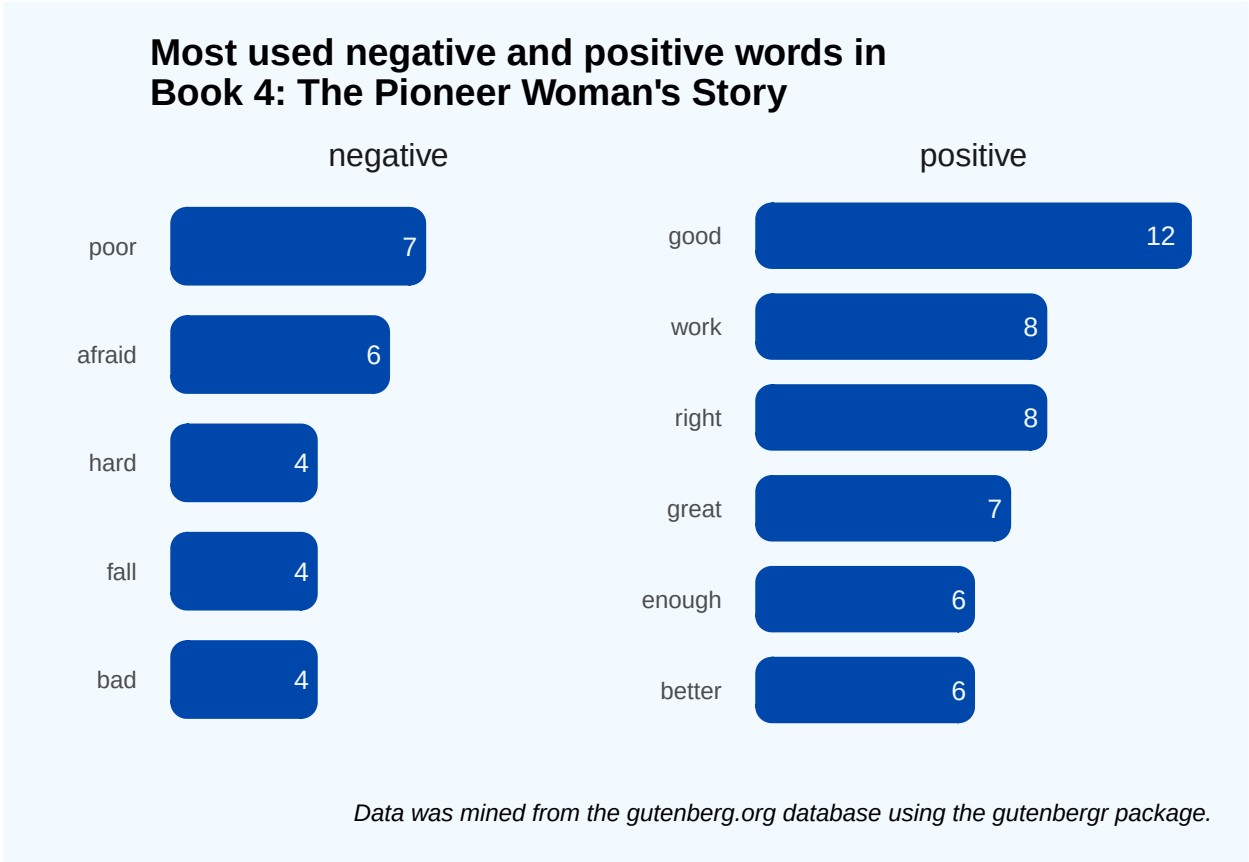

```
neg_pos_3<-df %>%
  filter(book_name=="Book 3: Lena Lingard") %>%
  unnest_tokens(word, text) %>%
  anti_join(get_stopwords()) %>%
  filter(!word=="like",
         !word=="burden",
         !word=="well") %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = T) %>%
  # ungroup() %>%
```

```
  group_by(sentiment) %>%
  # head(10) %>%
  top_n(5) %>%
  ungroup() %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(word, n, fill = sentiment)) +
  geom_chicklet(width = 0.7,
                radius=unit(2,"mm"),
                fill="#0047ab", color="#0047ab") +
  facet_wrap(~sentiment, scales = "free_y") +
  coord_flip() +
  my_theme() +
  geom_text(aes(label=n), hjust=1.5, color="#f3faff", size=3.5) +
  labs(x="", y="",
       caption = "Data was mined from the gutenberg.org database using the gutenbergr package.") +
    theme(axis.ticks.x=element_blank(),
          axis.ticks.y=element_blank(),
          axis.text.x=element_blank(),
          plot.caption=element_text(face="italic")) +
  ggtitle("Most used negative and positive words in\nBook 3: Lena Lingard")
neg_pos_3
```

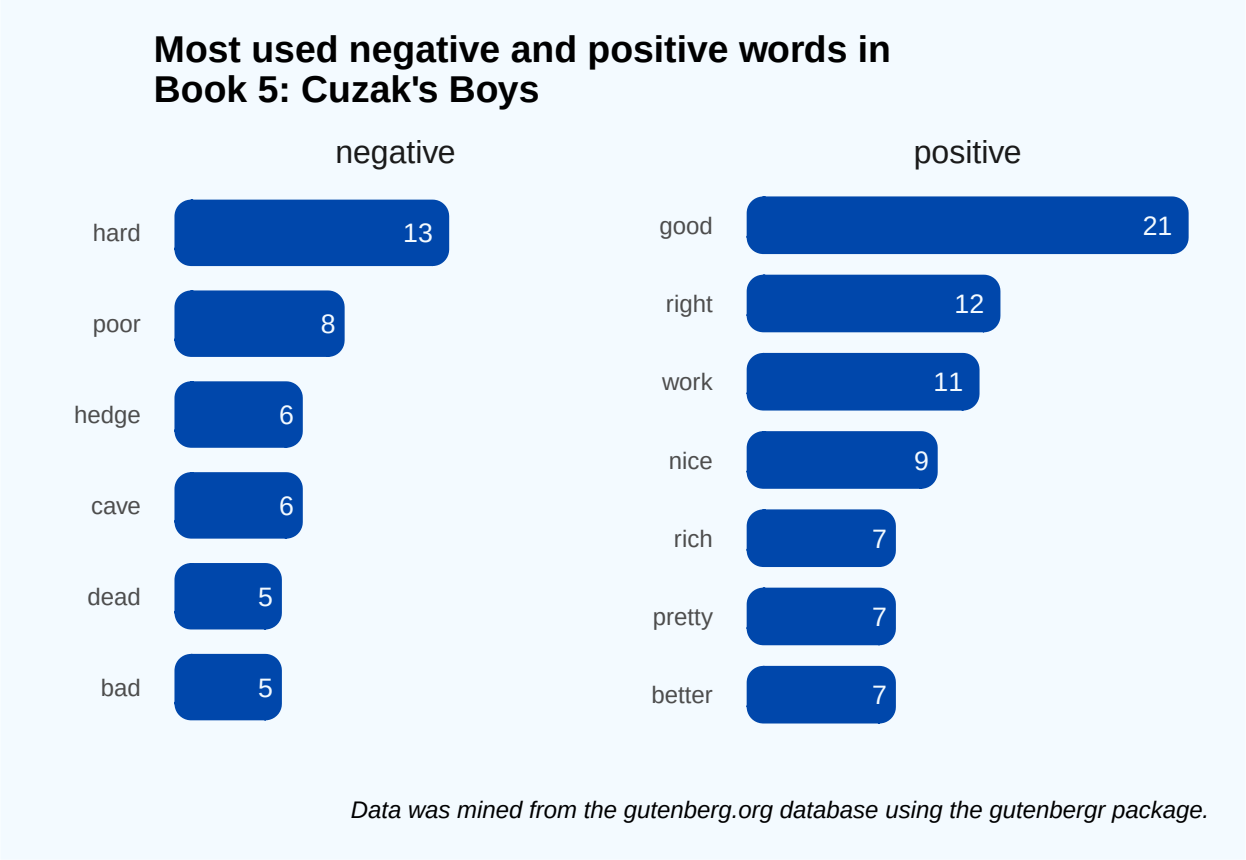



```
neg_pos_4<-df %>%
  filter(book_name=="Book 4: The Pioneer Woman's Story") %>%
  unnest_tokens(word, text) %>%
  anti_join(get_stopwords()) %>%
  filter(!word=="like",
         !word=="burden",
```

```
        !word=="well") %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = T) %>%
  # ungroup() %>%
  group_by(sentiment) %>%
  # head(10) %>%
  top_n(5) %>%
  ungroup() %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(word, n, fill = sentiment)) +
  geom_chicklet(width = 0.7,
                radius=unit(2,"mm"),
                fill="#0047ab", color="#0047ab") +
  facet_wrap(~sentiment, scales = "free_y") +
  coord_flip() +
  my_theme() +
  geom_text(aes(label=n), hjust=1.5, color="#f3faff", size=3.5) +
  labs(x="", y="",
       caption = "Data was mined from the gutenberg.org database using the gutenbergr package.") +
    theme(axis.ticks.x=element_blank(),
          axis.ticks.y=element_blank(),
          axis.text.x=element_blank(),
          plot.caption=element_text(face="italic")) +
  ggtitle("Most used negative and positive words in\nBook 4: The Pioneer Woman's Story")
neg_pos_4
```



```
neg_pos_5<-df %>%
  filter(book_name=="Book 5: Cuzak's Boys") %>%
  unnest_tokens(word, text) %>%
  anti_join(get_stopwords()) %>%
  filter(!word=="like",
         !word=="burden",
         !word=="well") %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = T) %>%
  # ungroup() %>%
  group_by(sentiment) %>%
  # head(10) %>%
  top_n(5) %>%
  ungroup() %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(word, n, fill = sentiment)) +
  geom_chicklet(width = 0.7,
                radius=unit(2,"mm"),
                fill="#0047ab", color="#0047ab") +
  facet_wrap(~sentiment, scales = "free_y") +
  coord_flip() +
  my_theme() +
  geom_text(aes(label=n), hjust=1.5, color="#f3faff", size=3.5) +
  labs(x="", y="",
       caption = "Data was mined from the gutenberg.org database using the gutenbergr package.") +
    theme(axis.ticks.x=element_blank(),
          axis.ticks.y=element_blank(),
          axis.text.x=element_blank(),
          plot.caption=element_text(face="italic")) +
  ggtitle("Most used negative and positive words in\nBook 5: Cuzak's Boys")
neg_pos_5
```



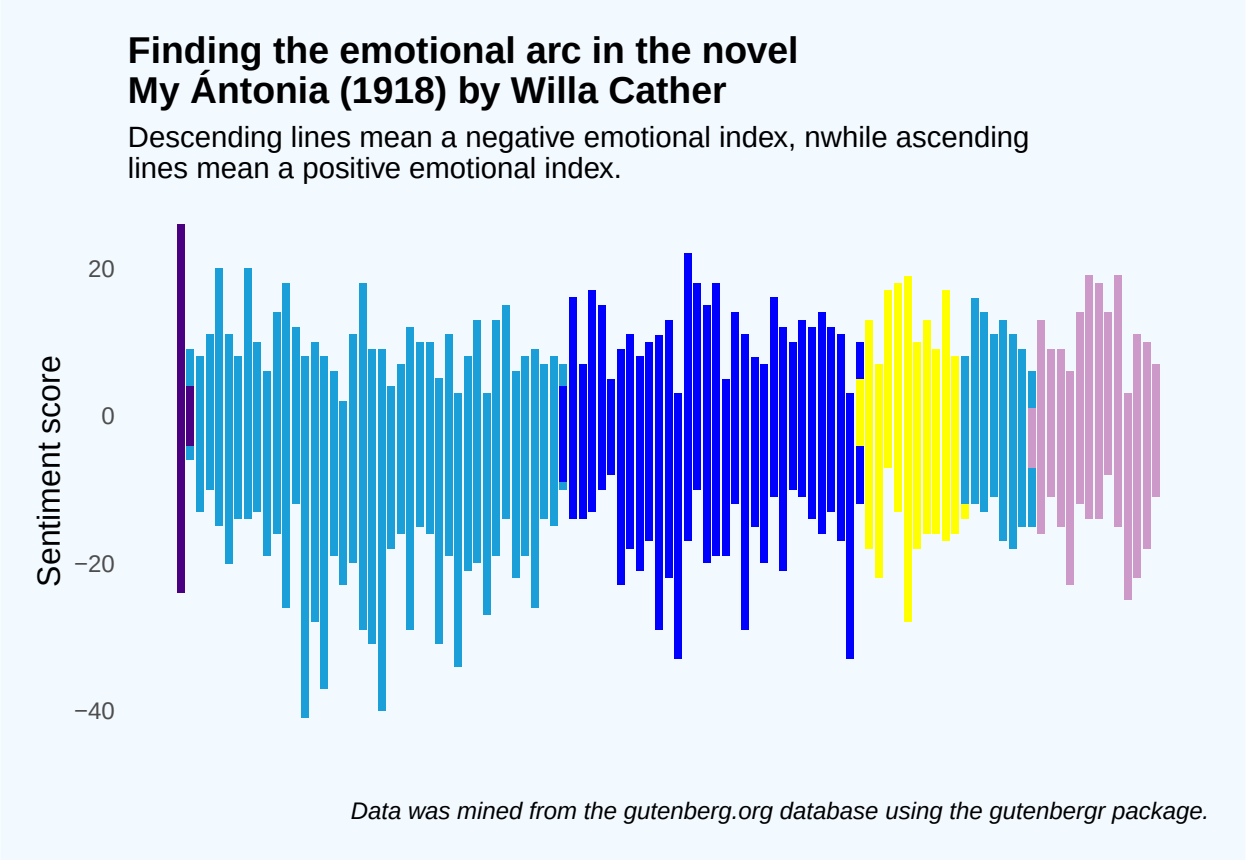
6.2 Finding the emotional arc of the novel

To find the emotional arc in the novel, where the rising action starts and ends, I must define an emotional index. An index is made my counting up the number of positive and negative up per book and compare this number with each other.

For example, if in a certain paragraph Jim recalls a certain story about Ambrosch he will use less favourable language than if he would be thinking about Lena. Therefore the emotional index of this paragraph will have a much lower number in the instance with Ambrosch than with Lena.

```
df %>%
  group_by(book_name, line_number) %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words) %>%
  inner_join(get_sentiments("bing")) %>%
  count(book_name, index = line_number %/% 80, sentiment) %>%
  pivot_wider(names_from = sentiment, values_from = n, values_fill = 0) %>%
  mutate(sentiment = positive - negative) %>%
  ggplot(aes(index, sentiment, fill = book_name)) +
  geom_col(width = 0.8) +
  ggtitle("Finding the emotional arc in the novel\nMy Ántonia (1918) by Willa Cather") +
  my_theme() + scale_fill_rcolorUtrecht(palette="hu") +
  labs(subtitle="Descending lines mean a negative emotional index, nwhile ascending\nlines mean a positi
       caption = "Data was mined from the gutenberg.org database using the gutenbergr package.",
       y="Sentiment score", x="") +
    theme(axis.ticks.x=element_blank(),
          axis.ticks.y=element_blank(),
```

```
        axis.text.x=element_blank(),
        plot.caption=element_text(face="italic"))
```



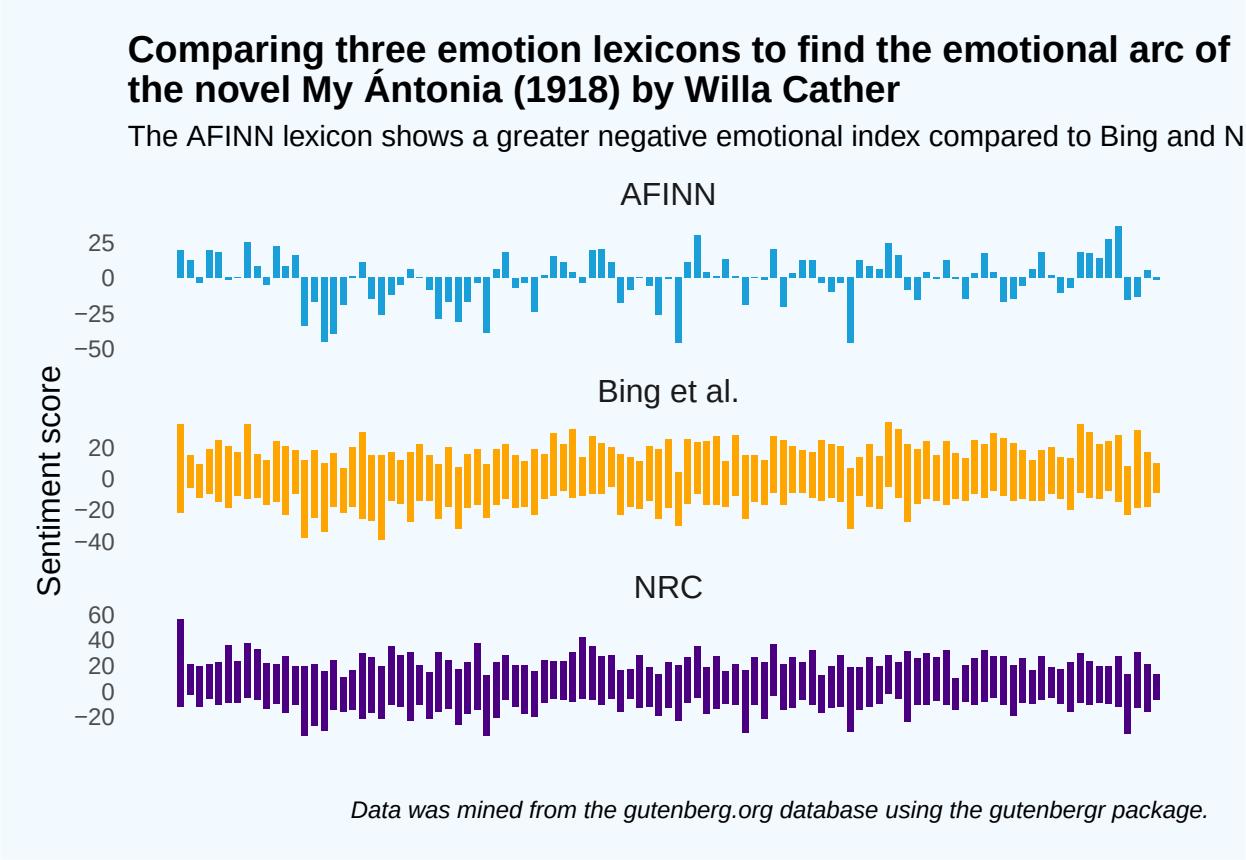
As seen in the graph, Book 1: The Shimerdas seen as light blue is the longest book in the novel but as it seems also the book that contains the biggest negative emotional index.

Above I mentioned four different emotion lexicons composed by different authors. These lexicons aren't very different from each other but may contain more negative words or more positive words. Hithero I have used only the Bing et al. lexicon which in sentiment analysis of text is most commonly used. Below I compare the results of the emotional index of the novel using the lexicons Bing et al., AFINN and NRC.

6.3 Comparing results of different emotion lexicons

```
# Source: https://www.tidytextmining.com/sentiment.html

afinn<-df %>%
  group_by(book_name, line_number) %>%
  unnest_tokens(output=word, input=text) %>%
  anti_join(stop_words, by="word") %>%
  inner_join(get_sentiments("afinn"), by="word") %>%
  group_by(index=line_number %/% 80) %>%
  summarize(sentiment=sum(value)) %>%
  mutate(method="AFINN")

bing_and_nrc<-bind_rows(
  df %>%
    group_by(book_name, line_number) %>%
```

```
    unnest_tokens(output=word, input=text) %>%
    inner_join(get_sentiments("bing")) %>%
    mutate(method = "Bing et al."),
  df %>%
    group_by(book_name, line_number) %>%
    unnest_tokens(output=word, input=text) %>%
    inner_join(get_sentiments("nrc") %>%
                 filter(sentiment %in% c("positive",
                                         "negative"))
    ) %>%
    mutate(method = "NRC")) %>%
  count(method, index = line_number %/% 80, sentiment) %>%
  pivot_wider(names_from = sentiment,
              values_from = n,
              values_fill = 0) %>%
  mutate(sentiment = positive - negative)

bind_rows(afinn,
          bing_and_nrc) %>%
  ggplot(aes(index, sentiment, fill = method)) +
  geom_col(show.legend = FALSE, width=0.7) +
  facet_wrap(~method, ncol = 1, scales = "free_y") +
  my_theme() + scale_fill_rcolorUtrecht(palette="hu") +
  labs(y="Sentiment score", x="",
       subtitle="The AFINN lexicon shows a greater negative emotional index compared to Bing and NRC.",
       caption = "Data was mined from the gutenberg.org database using the gutenbergr package.") +
  ggtitle("Comparing three emotion lexicons to find the emotional arc of\nthe novel My Ántonia (1918) by
      theme(axis.ticks.x=element_blank(),
          axis.ticks.y=element_blank(),
          axis.text.x=element_blank(),
          plot.caption=element_text(face="italic"))
```



Looking at the graph it seems as if the AFINN lexicon contains a higher number of words with a negative connotation. This way it looks as if the book is slightly more sad than it perhaps is.

6.4 Finding correlations between words

What is the first thing you think of when you hear the name Lena? This answer might differ depending on who is asked this question.

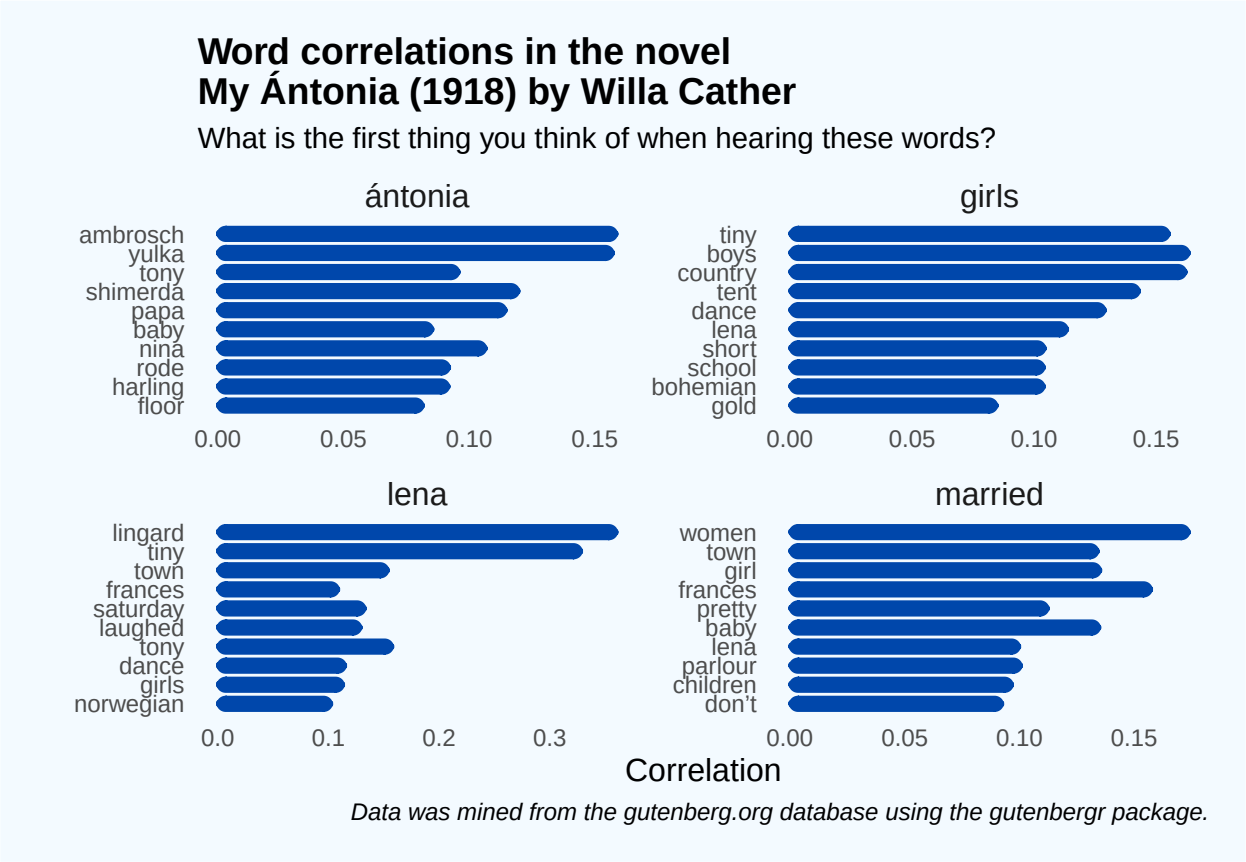
After splitting up the text into separate words, I can write code that counts the number of times each pair of words appear together within a section. This method applies the phi coefficient equation that gives a statistic estimate of how often a certain word is paired with our word of interest.

```
# Source: https://bookdown.org/Maxine/tidy-text-mining/counting-and-correlating-pairs-of-words-with-widy

df_section_words <- df %>%
  mutate(section = row_number() %/% 10) %>%
  filter(section > 0) %>%
  unnest_tokens(word, text) %>%
  filter(!word %in% stop_words$word)

library(widyr)
word_pairs <- df_section_words %>%
  pairwise_count(word, section, sort = TRUE)


word_cors <- df_section_words %>%
  group_by(word) %>%
  filter(n() >= 20) %>%
```

```
  pairwise_cor(word, section, sort = TRUE)


word_cors %>%
  filter(item1 == "ántonia")
```

```
## # A tibble: 269 x 3
##    item1   item2     correlation
##    <chr>   <chr>           <dbl>
##  1 ántonia ambrosch       0.159
##  2 ántonia yulka          0.157
##  3 ántonia shimerda       0.120
##  4 ántonia shimerdas      0.118
##  5 ántonia papa           0.115
##  6 ántonia nina           0.107
##  7 ántonia tony           0.0960
##  8 ántonia rode           0.0922
##  9 ántonia harling        0.0920
## 10 ántonia baby           0.0856
## # ... with 259 more rows
```

```
word_cors %>%
  filter(item1 %in% c("ántonia", "lena", "married", "girls")) %>%
  filter(!item1=="marry",
         !item2=="marry",
         !item2=="shimerdas",
         !item1=="she's",
         !item2=="she's",
         !item1=="didn't",
         !item2=="didn't") %>%
  group_by(item1) %>%
  slice_max(correlation, n = 10) %>%
  ungroup() %>%
  mutate(item2 = reorder(item2, correlation)) %>%
  ggplot(aes(item2, correlation)) +
    geom_chicklet(width = 0.7,
                  fill="#0047ab", color="#0047ab") +
  facet_wrap(~ item1, scales = "free") +
  coord_flip() + my_theme() +
  labs(x="", y="Correlation",
       subtitle="What is the first thing you think of when hearing these words?",
       caption = "Data was mined from the gutenberg.org database using the gutenbergr package.") +
  ggtitle("Word correlations in the novel\nMy Ántonia (1918) by Willa Cather") +
      theme(axis.ticks.x=element_blank(),
          axis.ticks.y=element_blank(),
          plot.caption=element_text(face="italic"))
```



Here we see that the word most correlated with the word *girls* is country. This may indicate that the Jim has tied the girls he meets in his youth to the plains where he grew up and from which he slowly detached when pursuing his career. The name Lena was surprisingly most correlated with her surname Lingard. The name Ántonia is most correlated with her brother's name Ambrosch and her sister Yulka, confirming her strong family ties.

7 Defining gender roles in the novel by correlating gender associated words

The novel My Ántonia was published in the year 1918 which is post-Gilded Age. Gender roles after this period changed a lot both in the North and the West of America.

A large part of the changing of gender roles on the prairie was in part due to The Great Migration when young women such as Ántonia, Lena and Tiny worked on the fields and sought other forms of employment to help sustain their family's well being.

By filtering gender pronouns from the novel I can find words that are most commonly paired with these pronouns which help in determining gender roles.

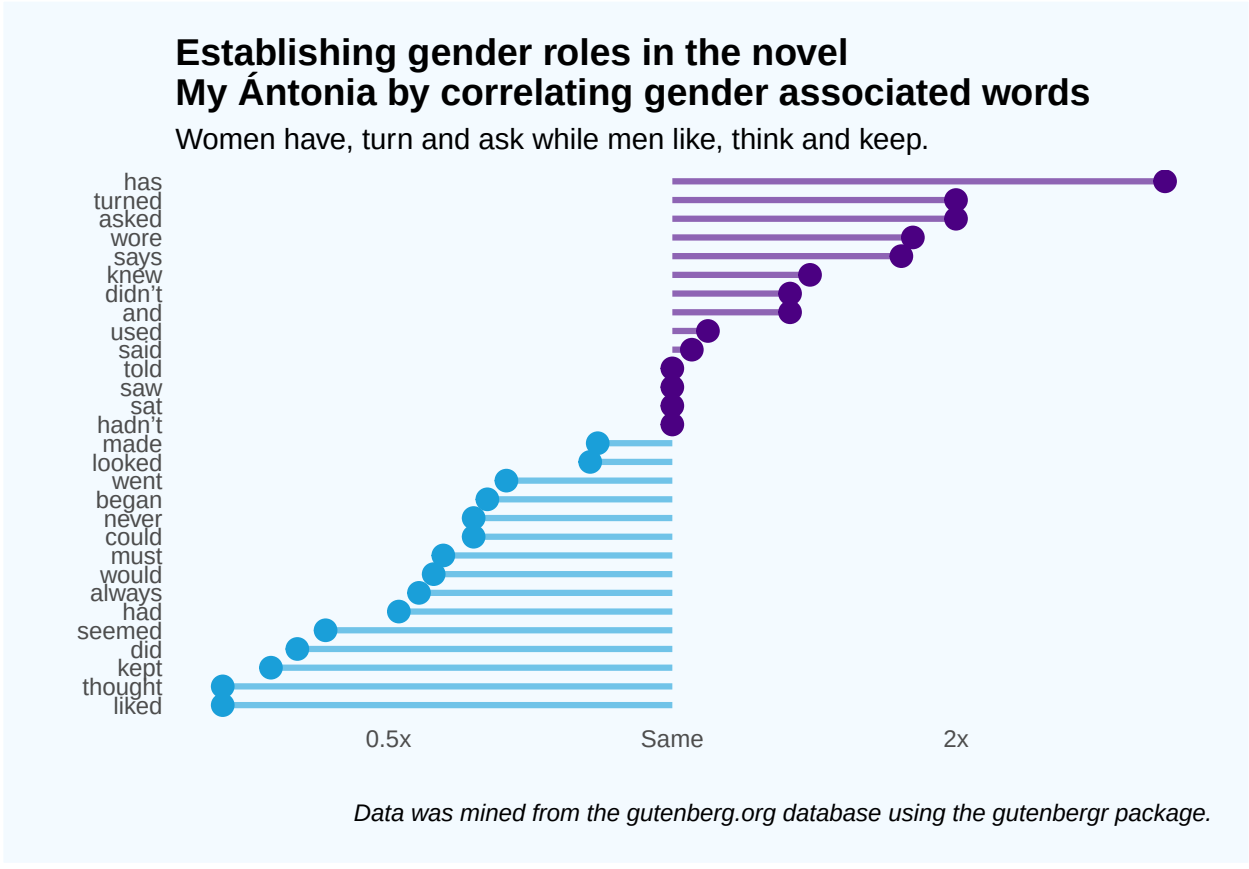
```
# Source: https://www.r-bloggers.com/2017/04/gender-roles-with-text-mining-and-n-grams/
bigrams <- df %>%
  unnest_tokens(bigram, text, token = "ngrams", n = 2)

bigrams_seperated <- bigrams %>%
  separate(bigram, c("word1", "word2"), sep = " ")

he_she_words <- bigrams_seperated %>%
  filter(word1 %in% c("he", "she"))
```

```
he_she_counts <- he_she_words %>%
  count(word1, word2) %>%
  spread(word1, n, fill = 0) %>%
  mutate(total = he + she,
         he = (he + 1) / sum(he + 1),
         she = (she + 1) / sum(she + 1),
         log_ratio = log2(she / he),
         abs_ratio = abs(log_ratio)) %>%
  arrange(desc(log_ratio))

he_she_words %>%
  count(word1, word2) %>%
  spread(word1, n, fill = 0) %>%
  mutate(total = he + she,
         he = (he + 1) / sum(he + 1),
         she = (she + 1) / sum(she + 1),
         log_ratio = log2(she/he),
         abs_ratio = abs(log_ratio)) %>%
  arrange(desc(log_ratio))
```

```
## # A tibble: 383 x 6
##    word2          he     she total log_ratio abs_ratio
##    <chr>       <dbl>   <dbl> <dbl>     <dbl>     <dbl>
##  1 leaned   0.000699 0.00558     6      3.00      3.00
##  2 laughed  0.000699 0.00398     4      2.51      2.51
##  3 seems    0.000699 0.00398     4      2.51      2.51
##  4 shook    0.000699 0.00398     4      2.51      2.51
##  5 can't    0.000699 0.00319     3      2.19      2.19
##  6 helped   0.000699 0.00319     3      2.19      2.19
##  7 murmured 0.000699 0.00319     3      2.19      2.19
##  8 thinks   0.000699 0.00319     3      2.19      2.19
##  9 became   0.000699 0.00239     2      1.77      1.77
## 10 danced   0.000699 0.00239     2      1.77      1.77
## # ... with 373 more rows
```

```
pronouns <- c("he", "she")

bigram_counts <- df %>%
  unnest_tokens(bigram, text, token = "ngrams", n = 2) %>%
  count(bigram, sort = T) %>%
  separate(bigram, c("word1", "word2"), sep = " ") %>%
  filter(word1 %in% pronouns) %>%
  count(word1, word2, wt = n, sort = T) %>%
  rename(total = n)

bigram_counts %>% group_by(word2) %>%
  filter(sum(total) > 10) %>%
  filter(!word2=="is",
         !word2=="was",
         !word2=="were") %>%
  ungroup() %>%
  spread(word1, total, fill = 0) %>%
  # mutate_if(is.numeric, funs((. + 1) / sum(. + 1))) %>%
  mutate(logratio = log2(she / he)) %>%
  arrange(desc(logratio)) %>%
```

```
  arrange(abs(logratio)) %>%
  mutate(abslogratio = abs(logratio)) %>%
  group_by(logratio < 0) %>%
  top_n(15, abslogratio) %>%
  ungroup() %>%
  mutate(word = reorder(word2, logratio)) %>%
  ggplot(aes(word, logratio, color = logratio <0)) +
  geom_segment(aes(x = word, xend = word,
                   y = 0, yend = logratio),
               size = 1.1, alpha = 0.6) +
  geom_point(size = 3.5) +
  coord_flip() + my_theme() +
  scale_color_discrete(name = "", labels=c("More 'she'", "More 'he'")) +
  scale_y_continuous(breaks = seq(-3, 3),
                     labels = c("0.125x", "0.25x", "0.5x", "Same", "2x", "4x", "8x")) +
  labs(title = "Establishing gender roles in the novel\nMy Ántonia by correlating gender associated word
       caption = "Data was mined from the gutenberg.org database using the gutenbergr package.",
       subtitle = "Women have, turn and ask while men like, think and keep.",
       y = "", x = "") +
      theme(axis.ticks.x=element_blank(),
          axis.ticks.y=element_blank(),
          plot.caption=element_text(face="italic")) +
  scale_color_rcolorUtrecht(palette = "hu")
```




```
summary<-left_join(total_sentences, total_words, by="book_name")
summary
```

```
## # A tibble: 6 x 3
```

```
##   book_name                          total_sentences total_word_count
##   <chr>                                        <int>            <int>
## 1 Book 1: The Shimerdas                         1949            31162
## 2 Book 2: The Hired Girls                       1604            24845
## 3 Book 3: Lena Lingard                           551             8373
## 4 Book 4: The Pioneer Woman's Story              390             6012
## 5 Book 5: Cuzak's Boys                           683            10363
## 6 Introduction                                    71             1341
```