Week 10 Assignment

MD SIMON CHOWDHURY

2024-11-8

Introduction

In this report, we explore sentiment analysis methods as discussed in Chapter 2 of Text Mining with R by Silge and Robinson (2017). Sentiment analysis allows us to quantify emotional tones within text, helping determine the overall sentiment—whether positive, negative, or neutral.

We will start by reproducing the primary example from Chapter 2 using Jane Austen's novels. Next, we will extend this analysis by applying it to a new corpus, Pride and Prejudice, and incorporating both the Bing and AFINN lexicons to capture a broader view of sentiment patterns.

Citation: Silge, J., & Robinson, D. (2017). Text Mining with R: A Tidy Approach. Chapter 2.

Loading and Preparing the Data

To begin, we will load Jane Austen's novels from the janeaustenr package and process them into a tidy text format, where each row represents a single word. This preparation step is essential for sentiment analysis.

```
# Load and tidy Jane Austen's novels
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)
```

Using the NRC Lexicon on Jane Austen's Novels

For the initial analysis, we will use the NRC lexicon to focus on words associated with "joy" in Jane Austen's Emma. This gives us a quick look at positive expressions within the text.

```
# Increase the download timeout to 300 seconds (5 minutes)
options(timeout = 300)

# Extract joy words from NRC lexicon
# Retrieve NRC lexicon from tidytext's built-in data
nrc_joy <- get_sentiments("nrc") %>%
  filter(sentiment == "joy")

# Find the most common joy words in "Emma"
tidy_books %>%
  filter(book == "Emma") %>%
  inner_join(nrc_joy) %>%
  count(word, sort = TRUE)
```

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

```
## # A tibble: 301 × 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
```

This step helps identify frequently used joy-related words like “good,” “friend,” and “happy,” offering insight into positive themes in Emma.

Exploring Sentiment Trajectory with the Bing Lexicon

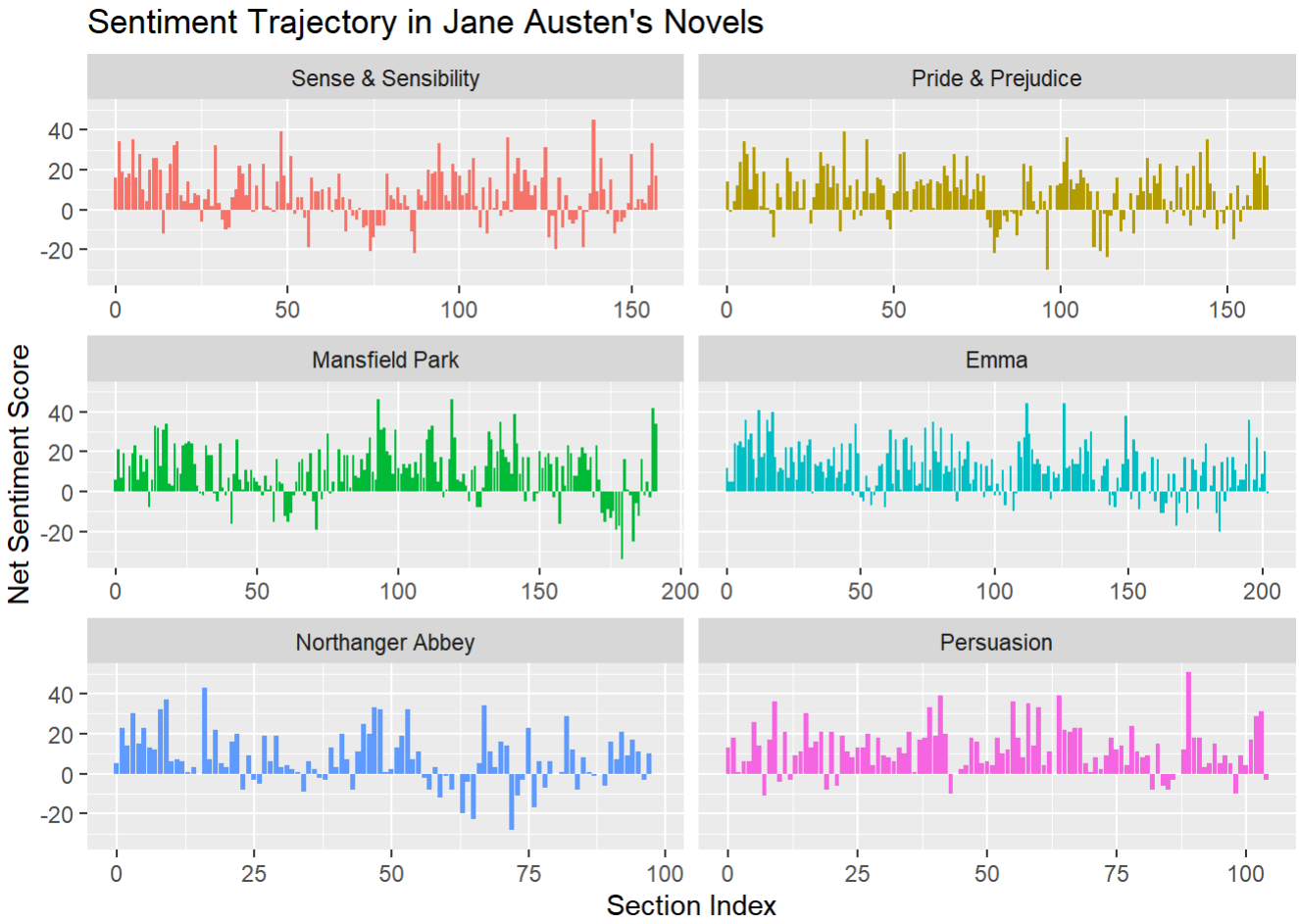
To visualize sentiment trends over the course of each novel, we’ll apply the Bing lexicon, which categorizes words as either positive or negative. By dividing each novel into chunks of 80 lines, we can observe shifts in sentiment over time.

```
# Calculate net sentiment for sections of each novel
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 rela
tionship between `x` and `y`.
## 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 sentiment trajectory
ggplot(jane_austen_sentiment, aes(index, sentiment, fill = book)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~book, ncol = 2, scales = "free_x") +
  labs(title = "Sentiment Trajectory in Jane Austen's Novels",
       x = "Section Index",
       y = "Net Sentiment Score")
```



This plot provides an overview of how sentiment changes over the narrative structure, with dips and peaks corresponding to positive or negative turns in the stories.

Extending the Analysis with Pride and Prejudice

To further explore sentiment analysis, we'll apply it to a new corpus: Pride and Prejudice from Project Gutenberg. This involves downloading the text and preparing it in the same tidy format.

```
file_path <- "https://raw.githubusercontent.com/simonchy/DATA607/refs/heads/main/week%2010/pr
ide%20and%20prejudice.txt"

# Read in the text
pride_text <- readLines(file_path, encoding = "UTF-8")

# Convert to a data frame for tidy processing
pride_df <- data.frame(text = pride_text, stringsAsFactors = FALSE) %>%
  mutate(line = row_number())

# Tidy the text into individual words and add a chapter indicator
tidy_pride <- pride_df %>%
  mutate(chapter = cumsum(str_detect(text, regex("^chapter ", ignore_case = TRUE)))) %>%
  unnest_tokens(word, text)
```

This code tidies the Pride and Prejudice text by tokenizing words and adding a chapter column.

Sentiment Analysis of Pride and Prejudice Using

Multiple Lexicons

Now, we calculate the sentiment trajectory for Pride and Prejudice using the Multiple Lexicons, dividing the text into chunks to observe sentiment fluctuations.

```
# Repeat of inner joins, group and summarise over lines by index for Bing lexicon
pride_bing <- tidy_pride %>%
  mutate(word_count = 1:n(),
         index = word_count %/% 500) %>%
  inner_join(get_sentiments("bing")) %>%
  count(index, sentiment) %>%
  spread(sentiment, n, fill = 0) %>%
  mutate(sentiment = positive - negative,
         method = "bing")
```

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

```
# Repeat for NRC lexicon
pride_nrc <- tidy_pride %>%
  mutate(word_count = 1:n(),
         index = word_count %/% 500) %>%
  inner_join(get_sentiments("nrc")) %>%
  count(index, sentiment) %>%
  spread(sentiment, n, fill = 0) %>%
  mutate(sentiment = positive - negative,
         method = "nrc")
```

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

```
## Warning in inner_join(., get_sentiments("nrc")): Detected an unexpected many-to-many relat
ionship between `x` and `y`.
## ℹ Row 6 of `x` matches multiple rows in `y`.
## ℹ Row 13027 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.
```

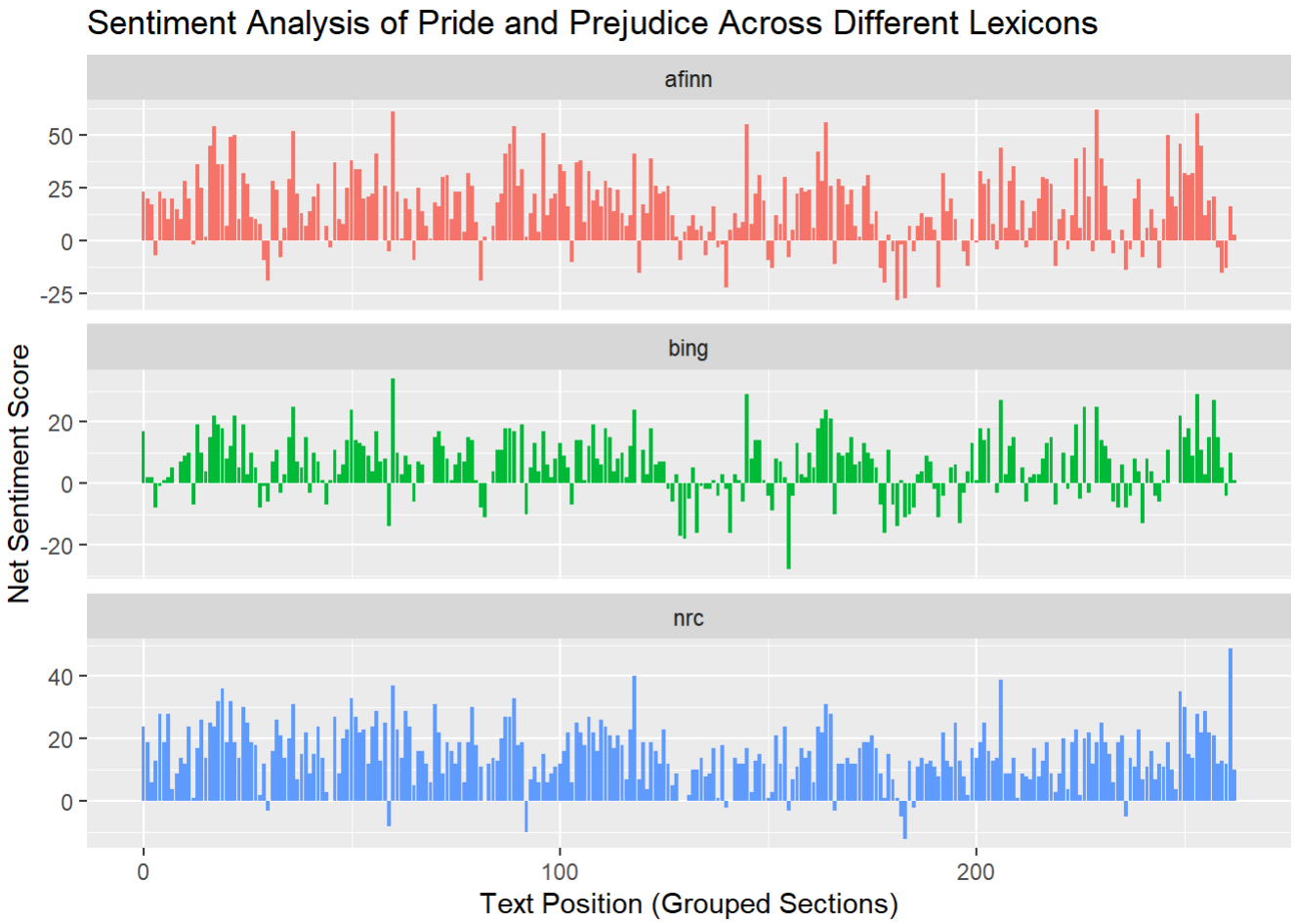
```
# Repeat for AFINN lexicon
pride_afinn <- tidy_pride %>%
  mutate(word_count = 1:n(),
         index = word_count %/% 500) %>%
  inner_join(get_sentiments("afinn")) %>%
  group_by(index) %>%
  summarise(sentiment = sum(value)) %>%
  mutate(method = "afinn")
```

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

```
# Create equal number of columns for binding
bing_pride <- subset(pride_bing, select = c("index", "sentiment", "method"))
nrc_pride <- subset(pride_nrc, select = c("index", "sentiment", "method"))
afinn_pride <- subset(pride_afinn, select = c("index", "sentiment", "method"))

# Bind them by rows with method as the identifier of lexicon
sentiment_comparison <- rbind(bing_pride, nrc_pride, afinn_pride)

# Visualize the distribution over the lines by lexicon type
ggplot(sentiment_comparison, aes(index, sentiment, fill = method)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~method, ncol = 1, scales = "free_y") +
  labs(title = "Sentiment Analysis of Pride and Prejudice Across Different Lexicons",
       x = "Text Position (Grouped Sections)",
       y = "Net Sentiment Score")
```



AFINN Lexicon: The AFINN lexicon shows a high level of variability with distinct peaks and troughs, reflecting intense positive and negative sentiment shifts throughout the text. This suggests that AFINN effectively captures moments of strong emotional expression, likely highlighting key events or conflicts in the narrative. Notably, some sections show sentiment scores as high as +50, indicating significant positivity, while others drop below -25, indicating notable negativity. This dynamic range makes AFINN suitable for identifying emotionally charged passages.

Bing Lexicon: The Bing lexicon's plot is less intense and smoother compared to AFINN. The sentiment values stay within a narrower range, with fewer drastic peaks and troughs, suggesting that Bing's binary positive/negative classification provides a more generalized sentiment overview. While it captures the general mood shifts across the narrative, it lacks the intensity observed with AFINN, making it useful for tracking broader sentiment trends rather than pinpointing specific moments of emotional intensity.

NRC Lexicon: The NRC lexicon shows a similar pattern to Bing but with slightly more variability. This indicates that NRC's broader vocabulary and emotional categories may allow it to capture subtle emotional shifts in the narrative. Although the sentiment range isn't as wide as AFINN's, NRC does provide more fluctuation than Bing, suggesting a balance between capturing general sentiment flow and some degree of emotional nuance.

Conclusion

In this report, we reproduced and extended sentiment analysis methods from Text Mining with R by applying them to both Jane Austen's novels and Pride and Prejudice. By using both the Bing and AFINN lexicons, we observed sentiment trends over time, capturing different facets of each story's emotional tone.

In summary, for Pride and Prejudice the AFINN lexicon provides the most dynamic range, capturing intense emotional highs and lows, while the Bing and NRC lexicons offer a more tempered and steady representation of sentiment. The NRC lexicon, in particular, offers a middle ground with moderate variability, which might be beneficial for analyzing subtle emotional tones in a literary text. Using multiple lexicons offers a comprehensive view of the text's emotional landscape, with each lexicon highlighting different aspects of sentiment in Pride and Prejudice.

These analyses highlight how sentiment lexicons can reveal underlying emotional currents in literature, providing a deeper understanding of the narrative structure and character dynamics.

Citation: Silge, J., & Robinson, D. (2017). Text Mining with R: A Tidy Approach. Chapter 2.