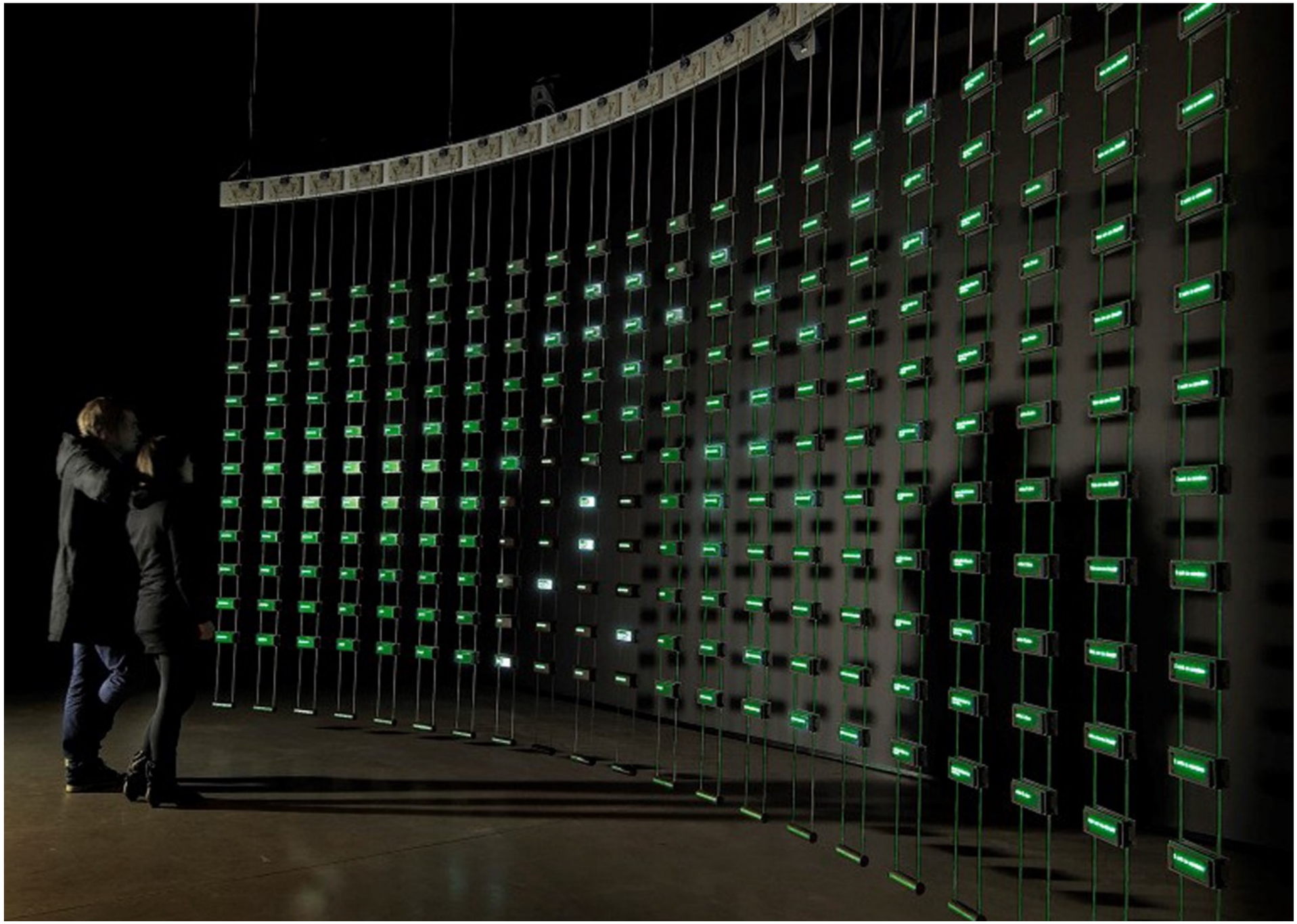
Text Data Analysis

Prof. Roger D. Peng
Department of Statistics and Data Sciences
University of Texas at Austin

Spring 2024

Text Data Art



https://youtu.be/Rzfnndd9fCk

Text Data Art

https://vimeo.com/benrubin/nytimes-art

Text Data

- Text can be converted into data using string processing and other tools
- Word counts are a typical data type
- Sentiment of words can be assessed using published lexicons
- Authorship styles based on word counts
- **tidytext** package designed to help with text analysis
- Large language models look at larger structures of language

Origin Story

Quantitative Analysis of Literary Styles

Roger D Peng & Nicolas W Hengartner

Pages 175-185 | Published online: 01 Jan 2012

Download citation https://doi.org/10.1198/000313002100

Supplemental Citations Metrics Reprints & Permissions Get access

Abstract

Writers are often viewed as having an inherent style that can serve as a literary fingerprint. By quantifying relevant features related to literary style, one may hope to classify written works and even attribute authorship to newly discovered texts. Beyond its intrinsic interest, the study of literary styles presents the opportunity to introduce and motivate many standard multivariate statistical techniques. Today the statistical analysis of literary styles is made much simpler by the wealth of real data readily available from the Internet. This article presents an overview and brief history of the analysis of literary styles. In addition we use canonical discriminant analyis and principal component analysis to identify structure in the data and distinguish authorship.

Text Data

Book I. The History Of A Family

Chapter I.
Fyodor Pavlovitch Karamazov

Alexey Fyodorovitch Karamazov was the third son of Fyodor Pavlovitch Karamazov, a land owner well known in our district in his own day, and still remembered among us owing to his gloomy and tragic death, which happened thirteen years ago, and which I shall describe in its proper place. For the present I will only say that this “landowner”—for so we used to call him, although he hardly spent a day of his life on his own estate—was a strange type, yet one pretty frequently to be met with, a type abject and vicious and at the same time senseless. But he was one of those senseless persons who are very well capable of looking after their worldly affairs, and, apparently, after nothing else. Fyodor Pavlovitch, for instance, began with next to nothing; his estate was of the smallest; he ran to dine at other men’s tables, and fastened on them as a toady, yet at his death it appeared that he had a hundred thousand roubles in hard cash. At the same time, he was all his life one of the most senseless, fantastical fellows in the whole district. I repeat, it was not stupidity—the majority of these fantastical fellows are shrewd and intelligent enough—but just senselessness, and a peculiar national form of it.

He was married twice, and had three sons, the eldest, Dmitri, by his first wife, and two, Ivan and Alexey, by his second. Fyodor Pavlovitch’s first wife, Adelaïda Ivanovna, belonged to a fairly rich and distinguished noble family, also landowners in our district, the Miüsovs. How it came to pass that an heiress, who was also a beauty, and moreover one of those vigorous, intelligent girls, so common in

Title: The Brothers Karamazov

Author: Fyodor Dostoyevsky

Translator: Constance Garnett

Release Date: February 12, 2009

Project Gutenberg

Sentiment Questions

- What is the general sentiment (positive or negative) in each of the book's chapters?

- How does the sentiment change from chapter to chapter?

Sentiment Analysis

- Read in the text
- Tokenize text into individual words
- Remove stop words
- Associate individual words with positive or negative sentiment
- Compute the proportion of positive sentiment words in each chapter
- Plot the positive sentiment across chapters

Reading the Text

```
brothers <- read_lines("pg28054.txt.gz")
```

```
> head(brothers, 20)
  [1] "The Project Gutenberg eBook of The Brothers Karamazov, by Fyodor Dostoyevsky"
  [2] ""
  [3] "This eBook is for the use of anyone anywhere in the United States and"
  [4] "most other parts of the world at no cost and with almost no restrictions"
  [5] "whatsoever. You may copy it, give it away or re-use it under the terms"
  [6] "of the Project Gutenberg License included with this eBook or online at"
  [7] "www.gutenberg.org. If you are not located in the United States, you"
  [8] "will have to check the laws of the country where you are located before"
  [9] "using this eBook."
 [10] ""
 [11] "Title: The Brothers Karamazov"
 [12] ""
 [13] "Author: Fyodor Dostoyevsky"
 [14] ""
 [15] "Translator: Constance Garnett"
 [16] ""
 [17] "Release Date: February 12, 2009 [eBook #28054]"
 [18] "[Most recently updated: January 22, 2023]"
 [19] ""
 [20] "Language: English"
```

Creating a Data Frame/Tibble

```
dat <- tibble(text = brothers)
```

Create a new data frame

A single column containing the lines of text

```
> dat
# A tibble: 37,628 × 1
   text
   <chr>
 1 "The Project Gutenberg eBook of The Brothers Karamazov, by Fyodor Dostoyevsky"
 2 ""
 3 "This eBook is for the use of anyone anywhere in the United States and"
 4 "most other parts of the world at no cost and with almost no restrictions"
 5 "whatsoever. You may copy it, give it away or re-use it under the terms"
 6 "of the Project Gutenberg License included with this eBook or online at"
 7 "www.gutenberg.org. If you are not located in the United States, you"
 8 "will have to check the laws of the country where you are located before"
 9 "using this eBook."
10 ""
# … with 37,618 more rows
# ℹ Use `print(n = ...)` to see more rows
```

Identifying Chapters

```
> dat |>
+     slice(166:182)
# A tibble: 17 × 1
   text
   <chr>
 1 "PART I"
 2 ""
 3 ""
 4 ""
 5 ""
 6 "Book I. The History Of A Family"
 7 ""
 8 ""
 9 ""
10 ""
11 "Chapter I."
12 "Fyodor Pavlovitch Karamazov"
13 ""
14 ""
15 "Alexey Fyodorovitch Karamazov was the third son of Fyodor Pavlovitch"
16 "Karamazov, a land owner well known in our district in his own day, and"
17 "still remembered among us owing to his gloomy and tragic death, which"
```

Identifying Chapters

```
dat |>
    mutate(chapter = str_detect(text, "^Chapter"),
           part = str_detect(text, "^PART")) |>
    slice(166:182)
```

Indicate lines of text that start with "Chapter"

Identifying Chapters

```
dat |>
    mutate(chapter = str_detect(text, "^Chapter"),
           part = str_detect(text, "^PART")) |>
    slice(166:182)
```

Indicate lines of text that start with "Chapter"

```
# A tibble: 17 × 3
   text                                                                                  chapter part
   <chr>                                                                                 <lgl>   <lgl>
 1 "PART I"                                                                              FALSE   TRUE
 2 ""                                                                                    FALSE   FALSE
 3 ""                                                                                    FALSE   FALSE
 4 ""                                                                                    FALSE   FALSE
 5 ""                                                                                    FALSE   FALSE
 6 "Book I. The History Of A Family"                                                     FALSE   FALSE
 7 ""                                                                                    FALSE   FALSE
 8 ""                                                                                    FALSE   FALSE
 9 ""                                                                                    FALSE   FALSE
10 ""                                                                                    FALSE   FALSE
11 "Chapter I."                                                                          TRUE    FALSE
12 "Fyodor Pavlovitch Karamazov"                                                         FALSE   FALSE
13 ""                                                                                    FALSE   FALSE
14 ""                                                                                    FALSE   FALSE
15 "Alexey Fyodorovitch Karamazov was the third son of Fyodor Pavlovitch"                FALSE   FALSE
16 "Karamazov, a land owner well known in our district in his own day, and"              FALSE   FALSE
17 "still remembered among us owing to his gloomy and tragic death, which"               FALSE   FALSE
>
```

Identifying Chapters

```
dat |>
    mutate(chapter = str_detect(text, "^Chapter"),
           part = str_detect(text, "^PART")) |>
    mutate(chapternum = cumsum(chapter)) |>
    slice(166:182)
```

Cumulative sum of the 'chapter' column

Identifying Chapters

```
dat |>
    mutate(chapter = str_detect(text, "^Chapter"),
           part = str_detect(text, "^PART")) |>
    mutate(chapternum = cumsum(chapter)) |>
    slice(166:182)
```

Cumulative sum of the 'chapter' column

```
# A tibble: 17 × 4
   text                                                                      chapter part  chapt…[1]
   <chr>                                                                     <lgl>   <lgl>   <int>
 1 "PART I"                                                                  FALSE   TRUE        0
 2 ""                                                                        FALSE   FALSE       0
 3 ""                                                                        FALSE   FALSE       0
 4 ""                                                                        FALSE   FALSE       0
 5 ""                                                                        FALSE   FALSE       0
 6 "Book I. The History Of A Family"                                         FALSE   FALSE       0
 7 ""                                                                        FALSE   FALSE       0
 8 ""                                                                        FALSE   FALSE       0
 9 ""                                                                        FALSE   FALSE       0
10 ""                                                                        FALSE   FALSE       0
11 "Chapter I."                                                              TRUE    FALSE       1
12 "Fyodor Pavlovitch Karamazov"                                             FALSE   FALSE       1
13 ""                                                                        FALSE   FALSE       1
14 ""                                                                        FALSE   FALSE       1
15 "Alexey Fyodorovitch Karamazov was the third son of Fyodor Pavlovitch"  FALSE   FALSE       1
16 "Karamazov, a land owner well known in our district in his own day, an… FALSE   FALSE       1
17 "still remembered among us owing to his gloomy and tragic death, which" FALSE   FALSE       1
# … with abbreviated variable name [1]chapternum
```

Identifying Blank Lines

```
dat |>
    mutate(chapter = str_detect(text, "^Chapter"),
           part = str_detect(text, "^PART")) |>
    mutate(chapternum = cumsum(chapter)) |>
    mutate(blank = str_detect(text, "[a-zA-Z0-9]+", negate = TRUE)) |>
    slice(166:182)
```

Find the opposite of a letter or number

Identifying Blank Lines

```
dat |>
    mutate(chapter = str_detect(text, "^Chapter"),
           part = str_detect(text, "^PART")) |>
    mutate(chapternum = cumsum(chapter)) |>
    mutate(blank = str_detect(text, "[a-zA-Z0-9]+", negate = TRUE)) |>
    slice(166:182)
```

Find the opposite of a letter or number

```
# A tibble: 17 × 5
   text                                                                  chapter part  chapt…[1] blank
   <chr>                                                                 <lgl>   <lgl>   <int> <lgl>
 1 "PART I"                                                              FALSE   TRUE        0 FALSE
 2 ""                                                                    FALSE   FALSE       0 TRUE
 3 ""                                                                    FALSE   FALSE       0 TRUE
 4 ""                                                                    FALSE   FALSE       0 TRUE
 5 ""                                                                    FALSE   FALSE       0 TRUE
 6 "Book I. The History Of A Family"                                     FALSE   FALSE       0 FALSE
 7 ""                                                                    FALSE   FALSE       0 TRUE
 8 ""                                                                    FALSE   FALSE       0 TRUE
 9 ""                                                                    FALSE   FALSE       0 TRUE
10 ""                                                                    FALSE   FALSE       0 TRUE
11 "Chapter I."                                                          TRUE    FALSE       1 FALSE
12 "Fyodor Pavlovitch Karamazov"                                         FALSE   FALSE       1 FALSE
13 ""                                                                    FALSE   FALSE       1 TRUE
14 ""                                                                    FALSE   FALSE       1 TRUE
15 "Alexey Fyodorovitch Karamazov was the third son of Fyodor Pavlo…     FALSE   FALSE       1 FALSE
16 "Karamazov, a land owner well known in our district in his own d…     FALSE   FALSE       1 FALSE
17 "still remembered among us owing to his gloomy and tragic death,…     FALSE   FALSE       1 FALSE
# … with abbreviated variable name [1]chapternum
```

Filtering

```
dat |>
    mutate(chapter = str_detect(text, "^Chapter"),
           part = str_detect(text, "^PART")) |>
    mutate(chapternum = cumsum(chapter)) |>
    mutate(blank = str_detect(text, "[a-zA-Z0-9]+", negate = TRUE)) |>
    filter(!part & !chapter & !blank & chapternum > 0)
```

```
# A tibble: 30,874 × 5
   text                                                      chapter part  chapt…[1] blank
   <chr>                                                     <lgl>   <lgl>   <int> <lgl>
 1 Fyodor Pavlovitch Karamazov                               FALSE   FALSE       1 FALSE
 2 Alexey Fyodorovitch Karamazov was the third son of Fyodor Pavlov… FALSE   FALSE       1 FALSE
 3 Karamazov, a land owner well known in our district in his own da… FALSE   FALSE       1 FALSE
 4 still remembered among us owing to his gloomy and tragic death, … FALSE   FALSE       1 FALSE
 5 happened thirteen years ago, and which I shall describe in its p… FALSE   FALSE       1 FALSE
 6 place. For the present I will only say that this “landowner”—for… FALSE   FALSE       1 FALSE
 7 used to call him, although he hardly spent a day of his life on … FALSE   FALSE       1 FALSE
 8 estate—was a strange type, yet one pretty frequently to be met w… FALSE   FALSE       1 FALSE
 9 type abject and vicious and at the same time senseless. But he w… FALSE   FALSE       1 FALSE
10 of those senseless persons who are very well capable of looking … FALSE   FALSE       1 FALSE
# … with 30,864 more rows, and abbreviated variable name [1]chapternum
# ℹ Use `print(n = ...)` to see more rows
```

Selecting

```
book <- dat |>
    mutate(chapter = str_detect(text, "^Chapter"),
           part = str_detect(text, "^PART")) |>
    mutate(chapternum = cumsum(chapter)) |>
    mutate(blank = str_detect(text, "[a-zA-Z0-9]+", negate = TRUE)) |>
    filter(!part & !chapter & !blank & chapternum > 0) |>
    select(-chapter, -part, -blank)
```

```
> book
# A tibble: 30,874 × 2
   text                                                                  chapternum
   <chr>                                                                      <int>
 1 Fyodor Pavlovitch Karamazov                                                    1
 2 Alexey Fyodorovitch Karamazov was the third son of Fyodor Pavlovitch           1
 3 Karamazov, a land owner well known in our district in his own day, and         1
 4 still remembered among us owing to his gloomy and tragic death, which          1
 5 happened thirteen years ago, and which I shall describe in its proper          1
 6 place. For the present I will only say that this “landowner”—for so we         1
 7 used to call him, although he hardly spent a day of his life on his own        1
 8 estate—was a strange type, yet one pretty frequently to be met with, a         1
 9 type abject and vicious and at the same time senseless. But he was one         1
10 of those senseless persons who are very well capable of looking after          1
# … with 30,864 more rows
# ℹ Use `print(n = ...)` to see more rows
```

Tokenization

New column name

Column in original data frame containing text lines

```
book_words <- book |>
    unnest_tokens(word, text, token = "words")
```

tidytext function for splitting lines of text into individual words

Tokenization

New column name

Column in original data frame containing text lines

```
book_words <- book |>
    unnest_tokens(word, text, token = "words")
```

tidytext function for splitting lines of text into individual words

```
> book_words
# A tibble: 354,673 × 2
   chapternum word
        <int> <chr>
 1          1 fyodor
 2          1 pavlovitch
 3          1 karamazov
 4          1 alexey
 5          1 fyodorovitch
 6          1 karamazov
 7          1 was
 8          1 the
 9          1 third
10          1 son
# … with 354,663 more rows
# ℹ Use `print(n = ...)` to see more rows
```

Stop Words

```
> stop_words
# A tibble: 1,149 × 2
   word        lexicon
   <chr>       <chr>
 1 a           SMART
 2 a's         SMART
 3 able        SMART
 4 about       SMART
 5 above       SMART
 6 according   SMART
 7 accordingly SMART
 8 across      SMART
 9 actually    SMART
10 after       SMART
```

```
stop_words |>
    select(lexicon) |>
    distinct()
```

```
# A tibble: 3 × 1
  lexicon
  <chr>
1 SMART
2 snowball
3 onix
```

Stop Words

```
stoplist <- stop_words |>
    filter(lexicon == "onix") |>
    distinct()
```

Some words are repeated (probably by accident) so remove them

```
> stoplist
# A tibble: 398 × 2
   word    lexicon
   <chr>   <chr>
 1 a       onix
 2 about   onix
 3 above   onix
 4 across  onix
 5 after   onix
 6 again   onix
 7 against onix
 8 all     onix
 9 almost  onix
10 alone   onix
```

Removing Stop Words

Original word data frame

```
> book_words
# A tibble: 354,673 × 2
   chapternum word
        <int> <chr>
 1          1 fyodor
 2          1 pavlovitch
 3          1 karamazov
 4          1 alexey
 5          1 fyodorovitch
 6          1 karamazov
 7          1 was
 8          1 the
 9          1 third
10          1 son
```

```
book_words |>
    anti_join(stoplist, by = "word")
```

```
# A tibble: 127,721 × 2
   chapternum word
        <int> <chr>
 1          1 fyodor
 2          1 pavlovitch
 3          1 karamazov
 4          1 alexey
 5          1 fyodorovitch
 6          1 karamazov
 7          1 third
 8          1 son
 9          1 fyodor
10          1 pavlovitch
```

Sentiment Data

```
book_words |>
    anti_join(stoplist, by = "word") |>
    inner_join(sentiments, by = "word")
```

Sentiments data (random sample)

```
# A tibble: 10 × 2
   word            sentiment
   <chr>           <chr>
 1 malady          negative
 2 recommendation  positive
 3 perplexed       negative
 4 subsidizes      positive
 5 ragged          negative
 6 balk            negative
 7 neatest         positive
 8 complaints      negative
 9 cannibal        negative
10 miraculous      positive
```

```
# A tibble: 22,288 × 3
   chapternum word      sentiment
        <int> <chr>     <chr>
 1          1 gloomy    negative
 2          1 tragic    negative
 3          1 death     negative
 4          1 proper    positive
 5          1 strange   negative
 6          1 pretty    positive
 7          1 vicious   negative
 8          1 senseless negative
 9          1 senseless negative
10          1 capable   positive
```

What about words with no sentiment?

Sentiment Data

```
book_words |>
    anti_join(stoplist, by = "word") |>
    inner_join(sentiments, by = "word") |>
    group_by(chapternum, sentiment) |>
    summarize(n = n(),
              .groups = "drop")
```

```
# A tibble: 192 × 3
   chapternum sentiment     n
        <int> <chr>     <int>
 1          1 negative     56
 2          1 positive     48
 3          2 negative     32
 4          2 positive     20
 5          3 negative     90
 6          3 positive     47
 7          4 negative    151
 8          4 positive     82
 9          5 negative    140
10          5 positive    131
```

Sentiment Data

```
book_words |>
    anti_join(stoplist, by = "word") |>
    inner_join(sentiments, by = "word") |>
    group_by(chapternum, sentiment) |>
    summarize(n = n(),
              .groups = "drop") |>
    pivot_wider(names_from = "sentiment",
                values_from = "n")
```

```
# A tibble: 192 × 3
   chapternum sentiment     n
        <int> <chr>     <int>
 1          1 negative     56
 2          1 positive     48
 3          2 negative     32
 4          2 positive     20
 5          3 negative     90
 6          3 positive     47
 7          4 negative    151
 8          4 positive     82
 9          5 negative    140
10          5 positive    131
```

Sentiment Data

```
book_words |>
    anti_join(stoplist, by = "word") |>
    inner_join(sentiments, by = "word") |>
    group_by(chapternum, sentiment) |>
    summarize(n = n(),
              .groups = "drop") |>
    pivot_wider(names_from = "sentiment",
                values_from = "n")
```

```
# A tibble: 192 × 3
   chapternum sentiment     n
        <int> <chr>     <int>
 1          1 negative     56
 2          1 positive     48
 3          2 negative     32
 4          2 positive     20
 5          3 negative     90
 6          3 positive     47
 7          4 negative    151
 8          4 positive     82
 9          5 negative    140
10          5 positive    131
```

```
# A tibble: 96 × 3
   chapternum negative positive
        <int>    <int>    <int>
 1          1       56       48
 2          2       32       20
 3          3       90       47
 4          4      151       82
 5          5      140      131
 6          6       40       46
 7          7      123      104
 8          8      118       78
 9          9      102      121
10         10      157       83
```

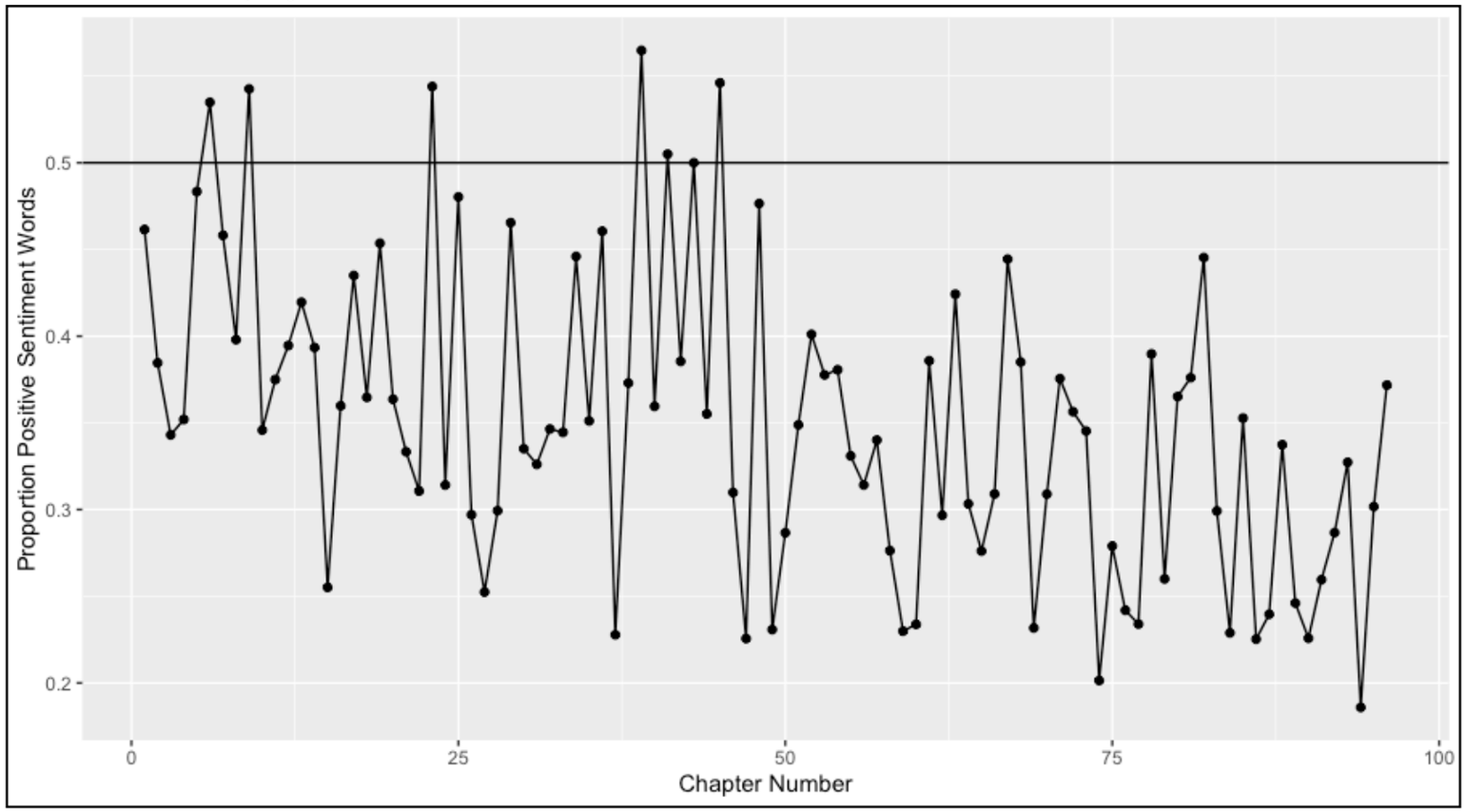
Sentiment Data

```
book_words |>
    anti_join(stoplist, by = "word") |>
    inner_join(sentiments, by = "word") |>
    group_by(chapternum, sentiment) |>
    summarize(n = n(),
              .groups = "drop") |>
    pivot_wider(names_from = "sentiment",
                values_from = "n") |>
  mutate(prop_positive = positive / (positive + negative))
```

```
# A tibble: 96 × 4
   chapternum negative positive prop_positive
        <int>    <int>    <int>         <dbl>
 1          1       56       48         0.462
 2          2       32       20         0.385
 3          3       90       47         0.343
 4          4      151       82         0.352
 5          5      140      131         0.483
 6          6       40       46         0.535
 7          7      123      104         0.458
 8          8      118       78         0.398
 9          9      102      121         0.543
10         10      157       83         0.346
```

Sentiment By Chapter

```
book_words |>
    anti_join(stoplist, by = "word") |>
    inner_join(sentiments, by = "word") |>
    group_by(chapternum, sentiment) |>
    summarize(n = n(),
              .groups = "drop") |>
    pivot_wider(names_from = "sentiment",
                values_from = "n") |>
    mutate(prop_positive = positive / (positive + negative)) |>
    ggplot(aes(chapternum, prop_positive)) +
    geom_point() +
    geom_line() +
    geom_hline(yintercept = 0.5) +
    labs(x = "Chapter Number", y = "Proportion Positive Sentiment Words")
```

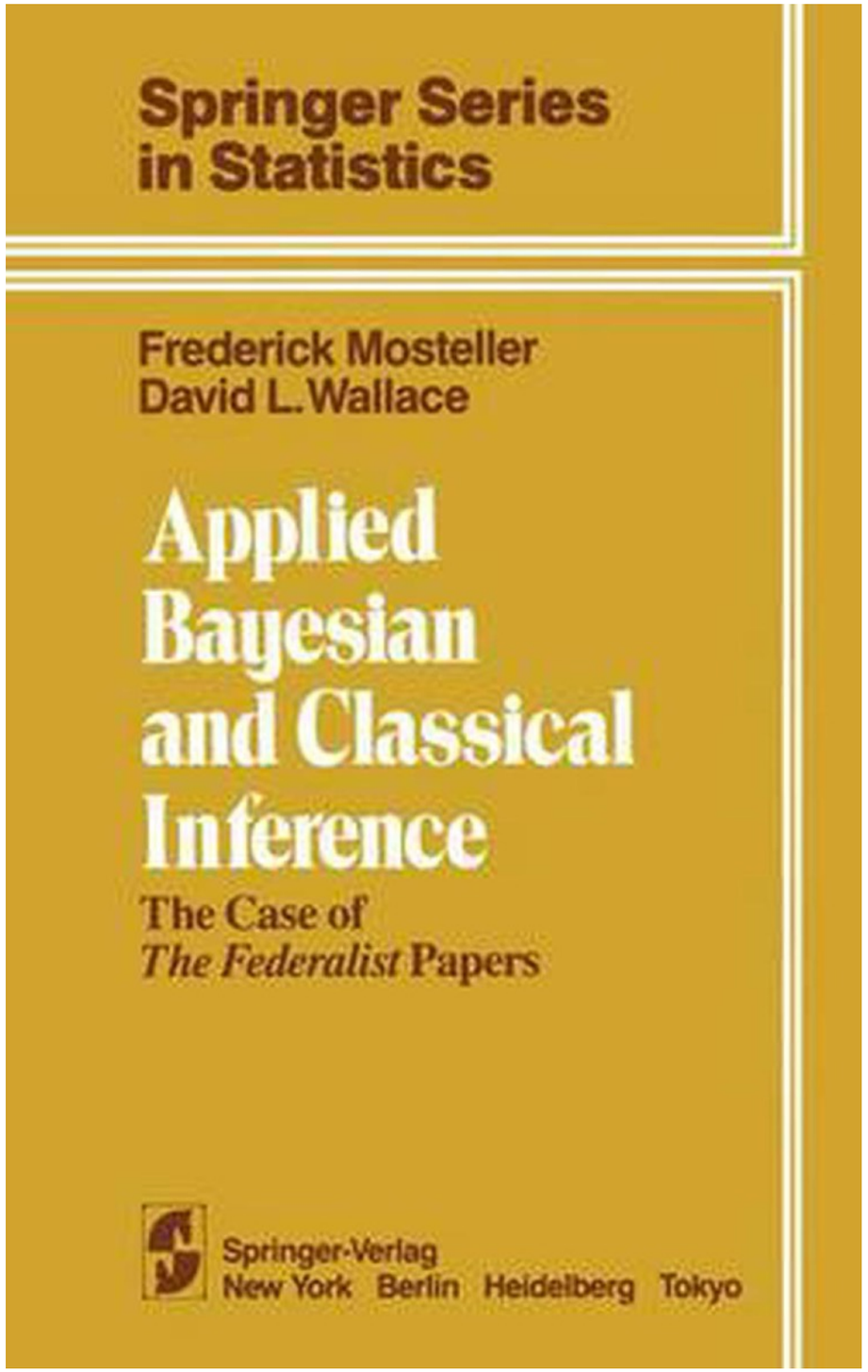


Sentiment Analysis Process

- Tokenize text into words
- Remove stop words using a standard list
- Join remaining words with a sentiment table/dictionary
- Compute a summary statistic of sentiment (i.e. proportion positive)
- Show sentiment by chapter or other unit of progression

Using Stop Words

- Stop words are sometimes considered to have "unconscious usage"
- Rates of usage may be an indicator of an author's unique style or writing pattern
- Sometimes called "function words"



Stop Word Analysis

- Divide the text into equal sized blocks of text (500 words)
- Remove words that are NOT stop words
- Count the occurrence of each stop word within each block of text (most will be 0)
- Compare mean rates of stop word usage between authors

Add Another Book

```
gatsby <- read_lines("pg64317.txt.gz")

dat <- bind_rows(
    tibble(text = brothers,
               book = "brothers karamazov"),
    tibble(text = gatsby,
           book = "great gatsby")
)
  dat
```

```
# A tibble: 44,400 × 2
   text                                                                    book
   <chr>                                                                   <chr>
 1 "The Project Gutenberg eBook of The Brothers Karamazov, by Fyodor Dostoyevsky" brothers karamazov
 2 ""                                                                      brothers karamazov
 3 "This eBook is for the use of anyone anywhere in the United States and"  brothers karamazov
 4 "most other parts of the world at no cost and with almost no restrictions"  brothers karamazov
 5 "whatsoever. You may copy it, give it away or re-use it under the terms"  brothers karamazov
 6 "of the Project Gutenberg License included with this eBook or online at"  brothers karamazov
 7 "www.gutenberg.org. If you are not located in the United States, you"  brothers karamazov
 8 "will have to check the laws of the country where you are located before"  brothers karamazov
 9 "using this eBook."                                                     brothers karamazov
10 ""                                                                      brothers karamazov
# ℹ 44,390 more rows
```

Tokenization

```
dat |>
    unnest_tokens(word, text, token = "words")
```

```
# A tibble: 407,574 × 2
   book                word
   <chr>               <chr>
 1 brothers karamazov the
 2 brothers karamazov project
 3 brothers karamazov gutenberg
 4 brothers karamazov ebook
 5 brothers karamazov of
 6 brothers karamazov the
 7 brothers karamazov brothers
 8 brothers karamazov karamazov
 9 brothers karamazov by
10 brothers karamazov fyodor
# ℹ 407,564 more rows
```

Ignore chapter numbers for this analysis

Create Text Blocks

```
dat |>
    unnest_tokens(word, text, token = "words") |>
    group_by(book) |>
    mutate(block = cut_interval(1:n(), length = 500, labels = FALSE)) |>
    ungroup()
```

```
# A tibble: 407,574 × 3
   book                  word       block
   <chr>                 <chr>      <int>
 1 brothers karamazov the                1
 2 brothers karamazov project            1
 3 brothers karamazov gutenberg          1
 4 brothers karamazov ebook              1
 5 brothers karamazov of                 1
 6 brothers karamazov the                1
 7 brothers karamazov brothers           1
 8 brothers karamazov karamazov          1
 9 brothers karamazov by                 1
10 brothers karamazov fyodor             1
# ℹ 407,564 more rows
```

Create block labels for each book

Join Stop Words

```
dat |>
    unnest_tokens(word, text, token = "words") |>
    group_by(book) |>
    mutate(block = cut_interval(1:n(), length = 500, labels = FALSE)) |>
    ungroup() |>
    inner_join(stoplist, by = "word")
```

```
# A tibble: 258,559 × 4
   book                 word  block lexicon
   <chr>                <chr> <int> <chr>
 1 brothers karamazov   the       1 onix
 2 brothers karamazov   of        1 onix
 3 brothers karamazov   the       1 onix
 4 brothers karamazov   by        1 onix
 5 brothers karamazov   this      1 onix
 6 brothers karamazov   is        1 onix
 7 brothers karamazov   for       1 onix
 8 brothers karamazov   the       1 onix
 9 brothers karamazov   use       1 onix
10 brothers karamazov   of        1 onix
# ℹ 258,549 more rows
```

Next:

1. **Group rows by book, word, and block.**
2. **Count the number of times each stop word occurs**

Count Stop Words

```
dat |>
    unnest_tokens(word, text, token = "words") |>
    group_by(book) |>
    mutate(block = cut_interval(1:n(), length = 500, labels = FALSE)) |>
    ungroup() |>
    inner_join(stoplist, by = "word") |>
    select(-lexicon) |>
    group_by(book, block, word) |>
    summarize(n = n(),
              .groups = "drop")
```

```
# A tibble: 81,856 × 4
   book                 block word          n
   <chr>                <int> <chr>     <int>
 1 brothers karamazov       1 a            16
 2 brothers karamazov       1 almost        1
 3 brothers karamazov       1 always        1
 4 brothers karamazov       1 an            2
 5 brothers karamazov       1 and           8
 6 brothers karamazov       1 another       1
 7 brothers karamazov       1 anyone        1
 8 brothers karamazov       1 anywhere      1
 9 brothers karamazov       1 are           2
10 brothers karamazov       1 at            5
# ℹ 81,846 more rows
```

Next:

1. **Group rows by book and word**
2. **Average the number of times each stop word occurs**

Average Across Blocks

```
dat |>
    unnest_tokens(word, text, token = "words") |>
    group_by(book) |>
    mutate(block = cut_interval(1:n(), length = 500, labels = FALSE)) |>
    ungroup() |>
    inner_join(stoplist, by = "word") |>
    select(-lexicon) |>
    group_by(book, block, word) |>
    summarize(n = n(),
              .groups = "drop") |>
    group_by(book, word) |>
    summarize(mean_word = mean(n),
              .groups = "drop") |>
    arrange(word)
```

```
# A tibble: 721 × 3
   book                 word   mean_word
   <chr>                <chr>      <dbl>
 1 brothers karamazov a            9.68
 2 great gatsby       a           14.1
 3 brothers karamazov about        1.83
 4 great gatsby       about        2.06
 5 brothers karamazov above        1.07
 6 great gatsby       above        1
 7 brothers karamazov across       1.08
 8 great gatsby       across       1.08
 9 brothers karamazov after        1.39
10 great gatsby       after        1.56
# ℹ 711 more rows
```

Next:

1. **Pivot to wide format in order to create separate columns for each book**
2. **Compute the difference in mean usage for each word**

Pivot Wider

```
dat |>
    unnest_tokens(word, text, token = "words") |>
    group_by(book) |>
    mutate(block = cut_interval(1:n(), length = 500, labels = FALSE)) |>
    ungroup() |>
    inner_join(stoplist, by = "word") |>
    select(-lexicon) |>
    group_by(book, block, word) |>
    summarize(n = n(),
              .groups = "drop") |>
    group_by(book, word) |>
    summarize(mean_word = mean(n),
              .groups = "drop") |>
  arrange(word) |>
  pivot_wider(names_from = "book",
              values_from = "mean_word") |>
  mutate(diff = `great gatsby` - `brothers karamazov`)
```

```
# A tibble: 379 × 4
   word    `brothers karamazov` `great gatsby`    diff
   <chr>                  <dbl>          <dbl>   <dbl>
 1 a                       9.68          14.1   4.45
 2 about                   1.83           2.06  0.236
 3 above                   1.07           1    -0.0685
 4 across                  1.08           1.08  0
 5 after                   1.39           1.56  0.173
 6 again                   1.72           1.30 -0.420
 7 against                 1.33           1.22 -0.111
 8 all                     3.39           2.63 -0.767
 9 almost                  1.35           1.26 -0.0849
10 alone                   1.24           1.31  0.0692
# ℹ 369 more rows
```

Next:

1. **Plot the difference in usage by word**

Plot Difference in Word Usage

```
dat |>
    unnest_tokens(word, text, token = "words") |>
    group_by(book) |>
    mutate(block = cut_interval(1:n(), length = 500, labels = FALSE)) |>
    ungroup() |>
    inner_join(stoplist, by = "word") |>
    select(-lexicon) |>
    group_by(book, block, word) |>
    summarize(n = n(),
              .groups = "drop") |>
    group_by(book, word) |>
    summarize(mean_word = mean(n),
              .groups = "drop") |>
    arrange(word) |>
    pivot_wider(names_from = "book",
                values_from = "mean_word") |>
    mutate(diff = `great gatsby` - `brothers karamazov`) |>
    ggplot(aes(x = 1:length(word), y = diff)) +
    geom_text(aes(label = word)) +
    labs(x = NULL, y = "Difference in Word Use (Gatsby - Brothers)")
```

Split text into words and create blocks of 500 words

Plot Difference in Word Usage

```
dat |>
    unnest_tokens(word, text, token = "words") |>
    group_by(book) |>
    mutate(block = cut_interval(1:n(), length = 500, labels = FALSE)) |>
    ungroup() |>
    inner_join(stoplist, by = "word") |>
    select(-lexicon) |>
    group_by(book, block, word) |>
    summarize(n = n(),
              .groups = "drop") |>
    group_by(book, word) |>
    summarize(mean_word = mean(n),
              .groups = "drop") |>
    arrange(word) |>
    pivot_wider(names_from = "book",
                values_from = "mean_word") |>
    mutate(diff = `great gatsby` - `brothers karamazov`) |>
    ggplot(aes(x = 1:length(word), y = diff)) +
    geom_text(aes(label = word)) +
    labs(x = NULL, y = "Difference in Word Use (Gatsby - Brothers)")
```

Join text with stop list to keep only stop words

Plot Difference in Word Usage

```
dat |>
    unnest_tokens(word, text, token = "words") |>
    group_by(book) |>
    mutate(block = cut_interval(1:n(), length = 500, labels = FALSE)) |>
    ungroup() |>
    inner_join(stoplist, by = "word") |>
    select(-lexicon) |>
    group_by(book, block, word) |>
    summarize(n = n(),
              .groups = "drop") |>
    group_by(book, word) |>
    summarize(mean_word = mean(n),
              .groups = "drop") |>
    arrange(word) |>
    pivot_wider(names_from = "book",
                values_from = "mean_word") |>
    mutate(diff = `great gatsby` - `brothers karamazov`) |>
    ggplot(aes(x = 1:length(word), y = diff)) +
    geom_text(aes(label = word)) +
    labs(x = NULL, y = "Difference in Word Use (Gatsby - Brothers)")
```

Count the occurrences of stop words in each book/block

Plot Difference in Word Usage

```
dat |>
    unnest_tokens(word, text, token = "words") |>
    group_by(book) |>
    mutate(block = cut_interval(1:n(), length = 500, labels = FALSE)) |>
    ungroup() |>
    inner_join(stoplist, by = "word") |>
    select(-lexicon) |>
    group_by(book, block, word) |>
    summarize(n = n(),
              .groups = "drop") |>
    group_by(book, word) |>
    summarize(mean_word = mean(n),
              .groups = "drop") |>
    arrange(word) |>
    pivot_wider(names_from = "book",
                values_from = "mean_word") |>
    mutate(diff = `great gatsby` - `brothers karamazov`) |>
    ggplot(aes(x = 1:length(word), y = diff)) +
    geom_text(aes(label = word)) +
    labs(x = NULL, y = "Difference in Word Use (Gatsby - Brothers)")
```

Compute the average use of each stop words in each book

Plot Difference in Word Usage

```
dat |>
    unnest_tokens(word, text, token = "words") |>
    group_by(book) |>
    mutate(block = cut_interval(1:n(), length = 500, labels = FALSE)) |>
    ungroup() |>
    inner_join(stoplist, by = "word") |>
    select(-lexicon) |>
    group_by(book, block, word) |>
    summarize(n = n(),
              .groups = "drop") |>
    group_by(book, word) |>
    summarize(mean_word = mean(n),
              .groups = "drop") |>
    arrange(word) |>
    pivot_wider(names_from = "book",
                values_from = "mean_word") |>
    mutate(diff = `great gatsby` - `brothers karamazov`) |>
    ggplot(aes(x = 1:length(word), y = diff)) +
    geom_text(aes(label = word)) +
    labs(x = NULL, y = "Difference in Word Use (Gatsby - Brothers)")
```

Compute the difference in average stop word usage

Plot Difference in Word Usage

```
dat |>
    unnest_tokens(word, text, token = "words") |>
    group_by(book) |>
    mutate(block = cut_interval(1:n(), length = 500, labels = FALSE)) |>
    ungroup() |>
    inner_join(stoplist, by = "word") |>
    select(-lexicon) |>
    group_by(book, block, word) |>
    summarize(n = n(),
              .groups = "drop") |>
    group_by(book, word) |>
    summarize(mean_word = mean(n),
              .groups = "drop") |>
    arrange(word) |>
    pivot_wider(names_from = "book",
                values_from = "mean_word") |>
    mutate(diff = `great gatsby` - `brothers karamazov`) |>
    ggplot(aes(x = 1:length(word), y = diff)) +
    geom_text(aes(label = word)) +
    labs(x = NULL, y = "Difference in Word Use (Gatsby - Brothers)")
```

Plot the difference in stop word usage by word

Plot Difference in Word Usage