Chapter 3: Investigating Speakers and Change Over Time Using Grouped Data

In previous chapters, we investigated how breaking up strings of text into individual words or multi-word phrases can give us insight into a corpus. We did things like count the number of words within an entire data frame. But historians often want to profile not merely the collective use of language, but also the role of specific speakers and how their language changed over time.

Understanding change over time begins with the question of what is to be understood over time. Is it a list of words referring to women? Is it an individual life? Do we care about the course of change over days, weeks, years, or decades – or within a single debate? For many historians, the answer might well be, "all of the above," which is fine. Counting one variable (say, references to women) by another variable of time (say, decades) is how we accomplish an analysis of change over time.

In this chapter, we will learn how to count over time – and the more general principle, which is analyzing one variable against another variable. We may count words by time, or words by speaker to find which speakers said these words the most.

In the language of the algorithm, understanding the association between variables is accomplished by "grouping" the data, allowing us to take other fields, like speaker name, year, and debate title, into account. This chapter will use these groupings to gain insight into the discourse of speakers in Parliament.

At this point, critical thinking about what we count is appropriate.The prospect of analyzing the individual writers in a corpus is an obvious way to understand the poles of debate. But when the subject of analysis is the parliament of Great Britain, one understandable reaction may be skepticism. Why study the language of individual speakers in Parliament when 19th-century British Parliament was mostly a small group of elite white men and not people who reflected the British population?

One answer is that parliamentary language is an index of social experience. We may come to the parliamentary debates interested in the experience of women or ethnicities; the research program here is neutral as to what the researcher wants to study.

Issues that rose to parliamentary attention were few. Ordinary people had to gather petitions to get parliament to entertain the vote for working people or women. When we look at parliament, we don't see all the labor that went into assembling the petition – but we do see a very important register of when discussions in parliament began to change. Using parliament as an index of political labor allows us to ask of any given subject, when did this subject merit institutional attention? Understanding the flow of power in parliament is a method useful to many researchers who are otherwise uninterested in identifying the characteristics of the "great men" of history.

In the exercises that follow, we anticipate that most researchers can see the value of finding out more about both the men who held powerful offices and the experience of women. We will therefore touch on both the mighty – including prime ministers and chancellors of the exchequer – and the vulnerable – showing how we can use data to navigate to the few women who testified in parliament. Careful navigation of data can teach us about both kinds of people.

Counting by Group to Find the Wordiest Speakers

Our first exercise is finding the most "wordy" Parliamentarians from 1830–that is, Parliamentarians who spoke the most words. Often, the Parliamentarians who spoke the most did so because fellow Parliamentarians trusted them to express the wants of their own social group.

To analyze speakers in Hansard, we will first need to import a new category of data from `hansardr`: `speaker_metadata`.

```
# load the hansardr library
library("hansardr")

# load the required data
data("speaker_metadata_1830")

head(speaker_metadata_1830)
```

```
##       sentence_id    speaker suggested_speaker ambiguous fuzzy_matched ignored
## 1 S3V0010P0_11508 Mr. Croker  john_croker_1539         0             0       0
## 2 S3V0010P0_11509 Mr. Croker  john_croker_1539         0             0       0
## 3 S3V0010P0_11510 Mr. Croker  john_croker_1539         0             0       0
## 4 S3V0010P0_11511 Mr. Croker  john_croker_1539         0             0       0
## 5 S3V0010P0_11512 Mr. Croker  john_croker_1539         0             0       0
## 6 S3V0010P0_11513 Mr. Croker  john_croker_1539         0             0       0
```

For now, we will just go over three relevant fields from `speaker_metadata`.

Each sentence from the Hansard corpus is assigned a unique ID, as represented by the `sentence_id` field. The speaker who stated a sentence is assigned the same ID. This allows us to join the data from speaker_metadata to other data from `hansardr`, such as the debate text.

The other two important fields we will address are the `speaker` and the `suggested_speaker` fields.

The `speaker` field contains the speaker name as it was originally transcribed within the Hansard corpus. The resulting field contains inconsistencies and a lack of standardization that can make analysis of speakers difficult. A single speaker may have been recorded by different permutations of his first, middle and last name(s) (for example, "William Gladstone" may be transcribed as "William E. Gladstone", "W. Gladstone", or "Mr. Gladstone," to name just a few). In other cases, a speaker may have been called by a rotating office title (like "Prime Minister").

It is also the case that different speakers are recorded by the same name. For instance, two people named Sir Robert Peel served in Parliament during the 1820s and both are evoked by the same spelling of the name; the older Sir Robert Peel (1750-1830), the industrialist, briefly overlapped with his son, Sir Robert Peel (1841-46), the future prime minister. Meanwhile, "Mr. W. Gladstone" could refer to William Ewart Gladstone or his son William Henry Gladstone.

To make analysis of speaker names more challenging yet, during the digitization processes, optical character recognition (OCR) errors were introduced into the speaker names. Common OCR error include interpreting a lower case "L" as an "i," or the lower case letter "a" as an "o."

Because of these nuances, any naive attempt to count words by speaker would be guaranteed to produce wrong results. It was paramount, therefore, that in managing the `hansardr` data, we produce an authoritative index of actual speakers that corresponded to the facts. In data science, adding a column with information inking this authoritative index to the names as presented is called "metadata."

The curators of the dataset – a team of historians and datascientists run by the authors – stored the metadata about the speakers in the "suggested_speaker" field. This field contains our suggestion for the true identity of the speaker. Importantly, this suggested_speaker field represents a marked improvement on existing data sources, tested by historians, but our data is not guaranteed for universal accuracy; hence it is only "suggested." In the suggested_speaker field, a unique id is given that refers to one unique speaker who was in parliament during the time covered by the database. While multiple speakers during the same period sometimes share the same name (like the two Gladstones or two Peels), each individual speaker is assigned a unique number. Thus, when we see the number 3104 in the suggested_speaker field, the curators of the database have made the inference that the W. Gladstone in question refers to William Gladstone, not his son William Henry.

In the following code, we will make our data set smaller and easier to work with by just selecting the "speaker," "suggested_speaker," and "text" fields before tokenizing the data into individual words.

```
library("tidytext")
library("tidyverse")
library("lubridate")

data("hansard_1830")
data("speaker_metadata_1830")

words_1830 <- hansard_1830 %>%
  left_join(speaker_metadata_1830) %>%
  select(speaker, suggested_speaker, text) %>%
  unnest_tokens(word, text)

head(words_1830)
```

```
##                 speaker suggested_speaker   word
## 1 The Duke of Buccleugh walter_scott_6566   rose
## 2 The Duke of Buccleugh walter_scott_6566     my
## 3 The Duke of Buccleugh walter_scott_6566  lords
## 4 The Duke of Buccleugh walter_scott_6566     in
## 5 The Duke of Buccleugh walter_scott_6566 rising
## 6 The Duke of Buccleugh walter_scott_6566     to
```

With a data frame containing fields for "speaker", "suggested_speaker", and "word," we are now in a position to perform a new kind of analysis to see the number of words each speaker contributed to Parliament using two new functions: `group_by()` and `summarize()`.

We use `group_by()` to organize the data by group before using summarize for our count. By using `group_by()` we can count the number of times one variable occurs, given the presence of another variable. We can count the number of words spoken by a speaker. We could also use group_by to count the words spoken in a given year – or by a given speaker in a given year, and so on.

The arguments taken by the command `group_by()` tell the computer which variables to associate with each other. To group by speaker, we make the argument of `group_by()` speaker, like so:

```
words_per_speaker_1830 <- words_1830 %>%
  group_by(speaker)
```

Alternatively, we could group_by suggested_speaker:

```
words_per_speaker_1830 <- words_1830 %>%
  group_by(suggested_speaker)
```

A useful function that often accompanies `group_by()` is `summarize()`, a command that tells the computer to apply a mathematical transformation to each of the groups created by group_by. Thus if we group_by(speaker) and then summarize(), we can count how many words each speaker spoke, or how many dates a speaker appeared on, or the first date when each speaker appeared.

The `summarize()` function can be used with any statistical transformation, for instance `mean()` or `max()`. Here, we will use `summarize()` with a function to count, which is `n()`. Finally, `arrange()` allows us to sort the speakers in descending order of total words spoken.

In the following code the argument `words_spoken = n()` tells `summarize()` to create a new column with the number of words that reflect each unique speaker

```
words_per_speaker_1830 <- words_1830 %>%
  group_by(speaker) %>%
  summarize(words_spoken = n()) %>%
  arrange(desc(words_spoken))

head(words_per_speaker_1830)
```

```
## # A tibble: 6 x 2
##   speaker                          words_spoken
##   <chr>                                   <int>
## 1 Mr. Hume                               856835
## 2 Mr. O'Connell                          823600
## 3 Sir Robert Peel                        822825
## 4 Lord John Russell                      782439
## 5 The Chancellor of the Exchequer        620830
## 6 Lord Brougham                          596096
```

Most students of British history will be familiar with the names of the speakers on this list, which include two prime ministers as well as one noted Irish orator and nationalist, Daniel O'Connell. The count above also shows us that among the most prolific speakers was the Chancellor of the Exchequer. How should we interpret this fact?

A naive analyst might assume that the Chancellor of the Exchequer was one person, the fifth most prolific speaker in the 1830s. Yet this would be an error of interpretation. The number of words spoken by the Chancellor counts the speeches of several distinct individuals.

The careful analyst will be concerned about the speaker classified as "The Chancellor of the Exchequer," for many individuals held this post over the course of the years 1830-39, which we are looking at now. The Chancellor of the Exchequer is the head financial officer of the United Kingdom.

A more experienced researcher might begin by using the suggested_speaker field – not the speaker field – to count words spoken by each individual. Notice that the lines of code below differ from the lines of code above only in that the argument of the function `group_by()` has been changed from "speaker" to "suggested_speaker:"

```
words_per_speaker_1830 <- words_1830 %>%
  filter(suggested_speaker != "") %>%
  group_by(suggested_speaker) %>%
  summarize(words_spoken = n()) %>%
  arrange(desc(words_spoken))

head(words_per_speaker_1830)
```

```
## # A tibble: 6 x 2
##   suggested_speaker    words_spoken
##   <chr>                       <int>
## 1 robert_peel_1664          1224576
## 2 henry_brougham_1679       1180672
## 3 joseph_hume_1712           875951
## 4 daniel_oconnell_2552       808577
## 5 john_russell_1885          802418
## 6 thomas_rice_2286           647551
```

In this view, we see the first and last names of individual speakers as well as their ID number from parliament (that is, Robert Peel is id 1664; 1664 is not, in this case, a date).

Nevertheless, our first pass revealed something interesting – as byways in the archives often do. It is interesting that Hansard represents the Chancellor of the Exchequer speaking more words than any other office – including that of Prime Minister – at least for the 1830s. The Chancellor of the Exchequer oversees the work of the Treasury. In many cases, the Chancellor of the Exchequer was a preliminary step to becoming Prime Minister. Why was the Chancellor of the Exchequer so important? And what were the various Chancellors talking about that required so much speechifying?

A Research Question: What were the Chancellors of the Exchequer talking about?

It may seem arbitrary to investigate the Chancellor of the Exchequer, but the office provides a perfect opportunity to test how the metadata indexing individual speakers works. Next, will use the suggested_speaker field to pull apart the many individuals associated with the office.

Because the dataset includes metadata about each speaker and how they are described, we can find the many individuals who occupied this office, and we can do research on them instead. In a sense, it's a happy accident of our research process that we used group_by() applied to the speaker field rather than the

authoritative suggested_speaker field, because the speaker field tells us about the way that individuals were actually designated in Hansard, including their office name.

Before we proceed, we should try to refine our research question to provide maximum historical insight so that we can start to imagine how our quantitative research process becomes worthwhile. For instance, we can ask questions about the diversity of views of the individuals who and we can ask questions about the office.

We would expect most of the Chancellor of the Exchequer's speeches in parliament to reflect a predominant concern with taxation and spending. However, it is also possible that different individuals who held this post prioritized different issues. How different were they, the Chancellors of the Exchequer for 1830-39?

To analyze the different speakers, the code that follows will use `group_by()` to group the speeches made by Chancellors of the Exchequer by suggested_speaker – the actual individuals behind the office. Then, we can inspect the number of words used by each individual Chancellor, comparing their favorite words.

A Custom Stop Words List

Before we proceed, a first step is to "clean" our data. In previous chapters we used an existing stop words list from `tidytext`. The `hansardr` library also provides its own stop words list.

It may be the case, however, that an analyst wants to curate their own stop words list that caters to their specific data set or research question. In the following code we create a custom stop words list by assigning a list of words to a tibble (a `tidyverse`-style data frame). In this case, suppose we want to restrict all words referring to parliament, ideas, or procedures; we only want to look at words that are more or less substantive about the subjects being discussed in parliament. s An appropriate stopwords list might look like this:

```
custom_stop_words = tibble(word = c("hon", "speaker", "house", "question", "lord", "bill",
"committee", "duty", "country", "time", "amount", "government", "proposed", "law",
"measure", "learned", "law", "sir", "respect", "public", "gentleman", "gentlemen",
"friend", "noble", "parliament", "expenditure", "revenue", "tax", "principle", "proposal",
"consideration", "duties", "stated", "parliament", "act", "opinion", "inquiry", "effect",
"subject", "object", "motion", "baronet", "crown", "propose", "estimates", "sum",
"account", "ministers", "majesty", "proposition", "persons", "principles", "service",
"found", "propositions", "office", "matter", "statement", "paid", "increase", "moved",
"means", "considerable", "supply", "intention", "debt", "received", "expense", "estimate",
"charges", "resolution", "notice", "report", "parties", "party", "surplus", "commons",
"parties", "class", "commission", "form", "answer", "commissioners", "appointment",
"officers", "saving", "appeared", "granted", "late", "day", "future", "opportunity",
"saving", "officers", "information", "extent", "authority", "session", "justice",
"believed", "support", "character", "carried", "plan", "paper", "clause",  "opposite",
"system", "argument", "rate", "considered", "bills", "increase", "result", "reference",
"prepared", "laid", "portion", "passed", "intended", "chancellor", "taxation", "classes",
"appointed", "established", "parish", "confidence", "evidence", "pensions", "bring",
"income", "connected", "referred", "objection", "hoped", "adopted", "chancellor",
"account", "ministers", "occasion", "reduction", "reductions", "service", "services",
"attention", "vote", "brought", "forward", "purpose", "wish", "called", "period",
"money", "purpose", "power", "charge", "view", "circumstances", "trade", "taxes", "list",
"civil", "wished", "people", "discussion", "applied", "proper", "taking", "amendment",
```

```
"statements", "repeal", "papers", "fund", "feeling", "measures", "minister", "purposes",
"payment", "stock", "increased", "bound", "grant", "grounds", "doubt", "applied", "stamp",
"manner", "conduct", "exchequer", "provide", "reduced", "table", "difficulties", "anxious",
"compared", "provide", "king", "individual", "expressed", "hear", "hope", "establishment",
"debate", "contrary", "instance", "introduced", "call", "lords", "existing", "half",
"head", "offices", "views", "leave", "respecting", "speech", "feel", "excise", "effected",
"economy", "night", "price", "majority", "reason", "disposed", "held", "claims", "leave",
"required", "moment", "admitted", "ready", "words", "issue", "allowed", "true", "treasury",
"administration", "consent", "advantage", "business", "calculated", "enter", "fair",
"nature", "oppose", "reign", "honour", "mode", "reduce", "difficulty", "favour", "credit",
"adopt", "additional", "individuals", "mind", "sale", "times", "amounted", "deficiency",
"laws", "til", "agreed", "alluded", "mentioned", "meant", "heard", "feelings", "engaged",
"consolidated", "conclusion", "circulation", "begged", "calculations", "afford", "acts",
"acted", "giving", "pledge", "matters", "contract", "proceed", "determined", "satisfaction",
"declare", "giving", "spirits", "resolutions", "pursue", "opposition", "force","pursue",
"force", "opinions", "refer", "produce", "appeal", "pay", "terms", "private", "reform",
"observations", "entitles", "laws", "alteration", "due", "gallant", "joint", "undoubtedly",
"alluded", "benefit", "consequence", "perfectly", "trusted", "loan", "proceed", "cent",
"regard", "political", "royal", "carry", "importance", "proceedings", "aware", "rates",
"similar", "examination", "details", "told", "supposed", "sufficient", "person", "necessity",
"millions", "emoluments", "difference", "department", "till", "submit", "lay", "imposed",
"claim", "branches", "recommended", "privy", "personal", "select", "salary", "revenues",
"preceding", "practice", "move", "meet", "hand", "funds", "formed", "expenses",
"establishments", "entitled", "dimunition", "settlement", "sense", "arrangement", "address",
"sufficient", "local", "clergy", "relief", "necessity", "application", "claim",
"responsibility", "situation", "satisfactory", "ground", "revenues", "quarter", "change",
"founded", "secretary", "provided", "read", "commutation", "body", "provision", "practice",
"difference", "composition", "condition", "hands", "provision", "capital", "responsibility",
"press", "knowledge", "raised", "friends", "altogether", "meet", "hand", "free",
"difference", "claim", "necessity", "arguments", "left", "single", "founded", "arrangement",
"admit", "total", "told", "operation", "apply", "shown", "person", "decision", "actual",
"returns", "return", "restrictions", "national", "limited", "hundred", "existed", "council",
"charged", "annual", "allowances", "proceeding", "word", "existed", "fairly", "loss", "spirit",
"west", "sum"))
```

We are ready to filter and clean our data and explore the langauge of speakers called Chancellor of the Exchequer.

Counting the Words of Each Speaker

In the following code we first remove instances of John Spencer and Rigby Wason, who spoke so few words as Chancellor of the Exchequer that he contributes little to our project. Then we eliminate numbers, stop words, and custom stop words before counting each speaker's words with `group_by()` and `summarize()`.

```
chancellor_of_the_exchequeur <- words_1830 %>%
  filter(str_detect(speaker, "Exchequer"),
```

```
        str_detect(word, "[a-z]"),
        suggested_speaker != "john_spencer_1234",
        suggested_speaker != "rigby_wason_3024",
        suggested_speaker != "",
        suggested_speaker != "chancellor of the exchequerr") %>%
  anti_join(stop_words) %>%
  anti_join(custom_stop_words) %>%
  group_by(suggested_speaker, word) %>%
  summarize(n = n()) %>%
  top_n(35) %>%
  ungroup()


chancellor_of_the_exchequeur %>%
  sample_n(5) %>%
  head()
```

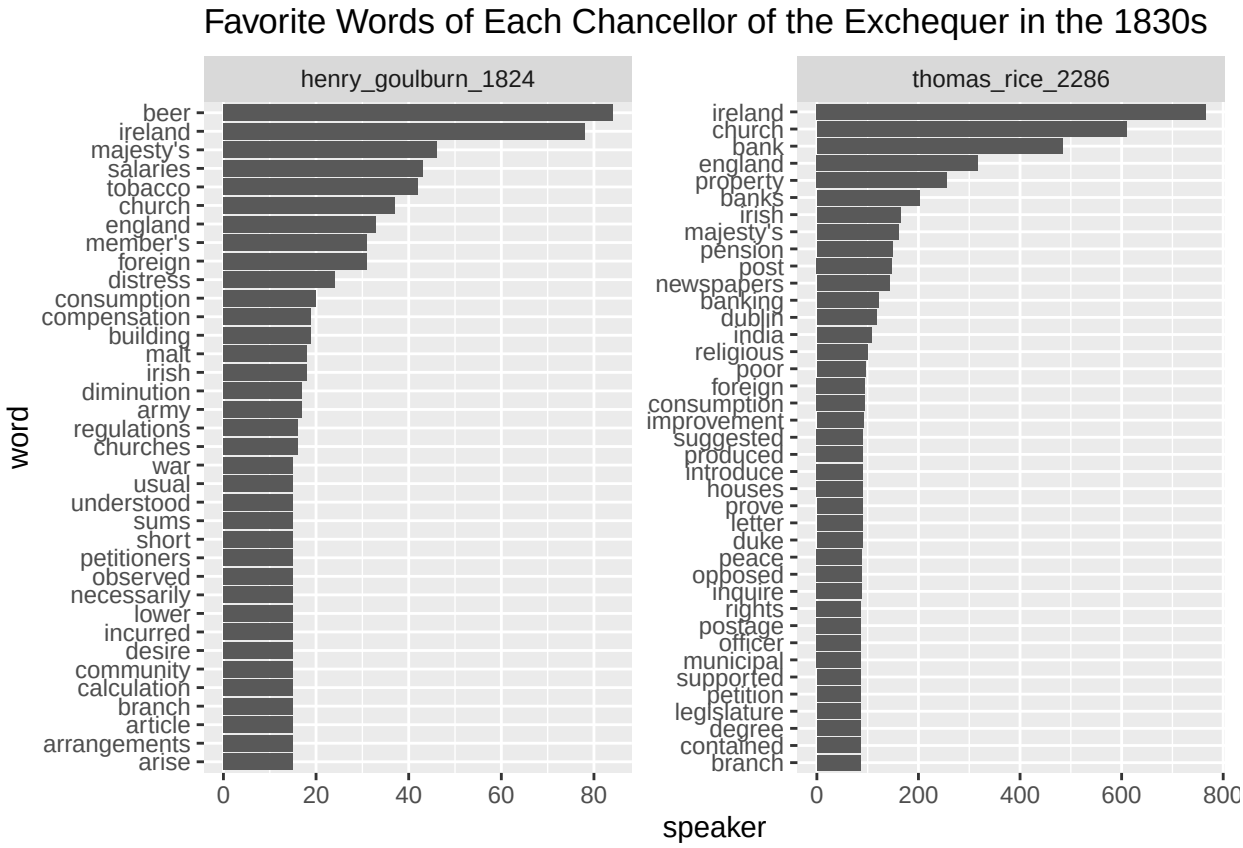
```
## # A tibble: 5 x 3
##   suggested_speaker   word             n
##   <chr>               <chr>        <int>
## 1 henry_goulburn_1824 article         15
## 2 henry_goulburn_1824 compensation    19
## 3 henry_goulburn_1824 community       15
## 4 henry_goulburn_1824 usual           15
## 5 thomas_rice_2286    consumption     95
```

Note that in the table above, the "word" is annotated with information about the speaker. This is to prevent confusion about what is being counted; we're not just counting how many times a word appears, but how many times each speaker said it.

Next, let's visualize the data so that we can see how many words are spoken.

```
chancellor_of_the_exchequeur <- chancellor_of_the_exchequeur %>%
  mutate(word = reorder_within(word, n, suggested_speaker))

ggplot(data = chancellor_of_the_exchequeur,
       aes(x = word, y = n)) +
  geom_col() +
  scale_x_reordered() +
  coord_flip() +
  facet_wrap(~suggested_speaker, scales = "free") +
  labs(x = "word", y = "speaker") +
  ggtitle("Favorite Words of Each Chancellor of the Exchequer in the 1830s")
```



The results show that the individuals who occupied the post of Chancellor of the Exchequer in fact represented a diversity of interests, ranging from Goulburn's interest in colonial commodities like "tobacco" to Rice's interest in the tension between the property and banking interests. Their territorial interests varied as wel Goulbourn emphasized England and Ireland, while Rice emphasized Ireland and India

Goulburn is interested in talking about commodities such as beer, tobacco, and malt; Peel is interested in addressing questions about how taxes relate to the church and its tithes, as well as political issues about how dissenters, Protestants and Catholics. Peel also thinks about commodities, but his commodities are different than Goulburn's; they include malt, land, and barley. Rice takes the issues in a different direction, reflecting on Dublin, India, banks, newspapers, and pensions. He shares with Peel a concern about property or land taxes.

Interpretation always represents an intellectual challenge. A list of most frequent words is not necessarily an index of importance: the most frequently-mentioned terms are typically the terms most subject to debate. Robert Peel, for instance, does not name Ireland's church tithes because he was interested in protecting the church in Ireland; to the contrary, he wished to abolish the system of tithes. Nor does the word list give us material for interpreting the rest of Peel's politics; the word list does nothing to tell the uninformed analyst about the movement for the abolition of taxes on grain, which forms a background to interpreting Peel's mentions of barley. The word list tells us little about the presence in 1830s Ireland of two churches, Roman Catholic and the Church of England. Without further reading, we might never known that Goulburn was famous for arguing for reductions on taxation (although with this knowledge, we might become interested

in the word "lower"). The analyst who comes to this data without a background in British history would do well to read in secondary sources before attempting to interpret the graph.

After some basic background research, however, the word lists can help produce insight, cluing the reader into differences such as Goulburn's relative willingness to talk about "distress," Peel's interest in "education," or Rice's interest in the post office. With word lists of this sort, we retrieve important clues about the salient differences contributed to political debates by individual ministers. A researcher interested in the careers of Chancellors of the Exchequer might start here before reading more deeply.

There is no rule of thumb for how many words to look at, but more words is always better. Distant reading, by its nature, is a shortcut to making sense of long passages of text. We want to take our time with these words, not rush to conclusions, and not merely read the top five. If we expanded the list of words, we might get more insight – especially if we have gone to the trouble of reading more about the work of Goulborn and Rice during the 1830s and we have specific questions about their priorities. The researcher is welcome to adjust upwards the number of words shown in these charts by raising the number in the code `top_n(35)` – although at a certain point, legibility will require new code to expand the bounding box that limits how big a visualization is.

How new is this information to readers of British history? We have long known that issues of what was taxed and how are a basic matter for politics of class, empire and nation; everyone wants someone else to pay for the government. The research modeled in this textbook has a pedagogical function, and for that reason, we often present in these chapters examples that could be improved or investigated further, and the above chart is one of those.

Critical Thinking About Our Results In any research process, it is important to reflect on how the choices we have made impact our analysis, and how making different choices might have produced different results. For some researchers, the list in `custom_stop_words` may unnecessarily eliminate words whose usage they might want to track. Some researchers will want to identify words for rhetorical statements such as "hoped" or "feeling"; others may be intrigued about how different speakers refer to "information" or "knowledge." In general, I have opted for a deep list which includes every general allusion to governance, institutions, discovery, and communication. The beauty of custom lists is that each researcher can easily adjust the words to match their interests Analysts should understand how carefully they must tailor a custom stop words list to their project, and how much of a difference additions and subtractions make. Using a custom stop word list is very much an artisinal skill of research; it involves much the same skills of judgment, awareness, curiosity, and sensitivity to background issues from theory and secondary sources as does careful reading of primary sources.

One complication of stopwords lists is that to produce functional results, a custom stop words list must often be long; the hundreds of performative, rhetorical, and governmental words listed below is not a perfect list, and it would not work for every exercise. In later chapters, we will investigate approaches such as differential measurement which allow the reader to skip over custom stop words lists to obtain information about what distinguishes one speaker from another or one year from the next. No process is perfect, however, and analysts must often resort to some kind of custom filter to get closer to information that is useful for their project. There is no silver bullet for creating an absolutely accurate tool where text-mining research always produces useful information; rather, iterative inquiry, paired with background reading and in-depth reading of primary sources, is the basis upon which all insight is ultimately made.

Working With Dates

Knowing who spoke the most in parliament is useful. But very often insight comes not from examining how counts change over time. What if we want to know not merely who spoke the most, but who was the top speaker for each year in parliament? Our `hansard_1830` data frame lists a date in the "speechdate" column, formatted as a four-digit year, two-digit month, and two digit day. To easily track speeches by year, we can add a new "field" – that is, a column – listing just the year of each speech.

Very frequently, when working with information about dates, we need to extract the month, day, or year from a data set. The `lubridate` package makes it easy to extract this information from dates in any format.

```
date1 <- mdy("6/24/1819")
date1
```

```
## [1] "1819-06-24"
```

```
year1 <- year(date1)
year1
```

```
## [1] 1819
```

We will use the `year()` function from `lubridate` with `mutate()` to create a new column that has just the extracted year from the `speechdate` field.

```
library("kableExtra")

data("debate_metadata_1830")

hansard_1830_w_year <- hansard_1830 %>%
  left_join(debate_metadata_1830) %>%
  mutate(year = year(speechdate))

# look at the first few columns of the dataset
hansard_1830_w_year %>%
  head() %>%
  mutate(text = str_trunc(text, 120)) %>%   # optional: shorten long lines
  kable("latex", booktabs = TRUE) %>%
  kable_styling(full_width = FALSE, position = "left") %>%
  column_spec(2, width = "4cm")    # wrap the "text" column
```

sentence_id	text	speechdate	debate
S2V0022P0_0	rose:— My Lords;—In rising to move that a humble Address be presented to his Majesty, in answer to the most gracious ...	1830-02-04	ADDRESS ON THE LORDS COMMISSIONERS SPEECH.]
S2V0022P0_1	It would be presumptuous in one so young and inexperienced as I am, and who have had the honour of a seat in your Hou...	1830-02-04	ADDRESS ON THE LORDS COMMISSIONERS SPEECH.]
S2V0022P0_2	I shall, therefore, confine myself to such few observations and reasons as may occur to me, claiming, at the same tim...	1830-02-04	ADDRESS ON THE LORDS COMMISSIONERS SPEECH.]
S2V0022P0_3	My lords, the strong assurances which his Majesty has been pleased to inform us he still continues to receive from th...	1830-02-04	ADDRESS ON THE LORDS COMMISSIONERS SPEECH.]
S2V0022P0_4	This nation is too much involved in the general interests of Europe not to view with satisfaction the intelligence th...	1830-02-04	ADDRESS ON THE LORDS COMMISSIONERS SPEECH.]
S2V0022P0_5	We cannot but hear, my lords, with satisfaction, of his Majesty's unremitting offices with his allies to carry into e...	1830-02-04	ADDRESS ON THE LORDS COMMISSIONERS SPEECH.]

Notice that the resulting dataset has a new field, "year."

If we want to count words by year, first we need to tokenize the speeches.

```
words_1830_w_year <- hansard_1830_w_year %>%
  left_join(speaker_metadata_1830) %>%
  select(speaker, suggested_speaker, text, year) %>%
  unnest_tokens(word, text)


words_1830_w_year %>%
  head() %>%
```

```
  select(speaker, word, year) %>%
  kable() #look at the first few columns of the dataset
```

speaker	word	year
The Duke of Buccleugh	rose	1830
The Duke of Buccleugh	my	1830
The Duke of Buccleugh	lords	1830
The Duke of Buccleugh	in	1830
The Duke of Buccleugh	rising	1830
The Duke of Buccleugh	to	1830

Now that we have a field for the year of each speech, we can return to "grouping" data to execute a faceted count where we count words by speaker and year. When we used the command `group_by()` above, we used it with one argument. But `group_by()` can take multiple arguments, allowing the analyst to perform grouped counts that involve multiple dimensions of the dataset. Next, we will use `group_by()` on two data fields at the same time – the "speaker" and "year" columns of as the arguments of `group_by()`, as in `group_by(speaker, year)`.

```
words_per_speaker_1830 <- words_1830_w_year %>%
  filter(suggested_speaker != "") %>%
  group_by(speaker, year) %>%
  summarize(speaker_words_per_year = n()) %>%
  arrange(desc(speaker_words_per_year))

head(words_per_speaker_1830)
```

```
## # A tibble: 6 x 3
## # Groups:   speaker [4]
##   speaker              year speaker_words_per_year
##   <chr>               <dbl>                  <int>
## 1 Lord Brougham        1839                 200436
## 2 Lord Althorp         1831                 180573
## 3 Lord Brougham        1838                 175070
## 4 The Lord Chancellor  1831                 161729
## 5 Mr. O'Connell        1834                 158617
## 6 Lord Althorp         1833                 154948
```

Notice that the result of grouping and counting is a data set with three columns: "speaker," "year," and "speaker_words_per_year." The last column is the count for each speaker per year – a field that makes it possible to analyze change over time.

When we use the `group_by()` command, the fields we use for grouping are preserved in the output. Two columns are the names of the facets we used for grouping – "speaker" and "year" – while the last column, "words_spoken," is the count of how many words are recorded for each speaker per year.

Women in 19th-century Parliament

For the most part, an analysis of speakers in Parliament will take the researcher down the road of the many elite white men who assumed powerful positions or passed honorary titles to their sons. Only 12 named women spoke in Parliament throughout the 19th-century. One woman who spoke the most was Mrs. Walrand from the 1824 debate, "Motion Respecting the Trial and Condemnation of Missionary Smith at Demerara." Exploring her role in this debate can give us insight and speculation into the criteria that had to be set in place for a woman to appear as a speaker and instrument to the power of Parliament.

"Motion Respecting the Trial and Condemnation of Missionary Smith at Demerara" reviews the trail, conviction, and death of John Smith, an Methodist missionary from England assigned to the British slave colony of Demerara in South America. Mr. Smith's trial accused him of assisting a slave rebellion that took place in Demerara in August of 1823. MP Brougham brought the topic of Mr. Smith's trial before Parliament to argue that the legal procedings convicting Mr. Smith were unjust and strayed from the guidance of Britain to the leaders of its Demerara colony. One could read his testimony as an early foundation for Britian's movement towards the Slavery Abolition Act of 1833, but his concerns–as well as the testimonies of other speakers–are nonetheless intertwined within a biased legal institution.

Mrs. Walrand spoke as a witness to the slave rebellion and served as a character witness condemning Mr. Smith. According to her testimony, she saw first hand the brutal violence of the rebells, having been a "defenceless lady" fired at by these "savages."

```
data("hansard_1820")
data("speaker_metadata_1820")
data("file_metadata_1820")

mrs_walrand_speaker_metadata <- speaker_metadata_1820 %>%
  filter(str_detect(speaker, regex("Mrs(.*)Walrand", ignore_case = T)))

mrs_walrand_sentences <- left_join(mrs_walrand_speaker_metadata, file_metadata_1820, by = "sentence_id")
  left_join(., hansard_1820, by = "sentence_id") %>%
  select(debate_id, text)

head(mrs_walrand_sentences) %>%
  kable()
```

debate_id	text
5142	proceeds: " "
5142	He (Mr. Forbes) said, how he envied Mr. Tucker his immediate death; and seemed in the most excruciating agony, but perfectly in his senses.
5142	I entreated the guard, in the name of every principle of humanity, just to let me send to Golden Grove, the next estate, to Dr. Goldie; I tried to get them to look at the dying, bleeding man, hoping the sight of his misery would move their compassion.
5142	Each of the guards, at different times, Murphy, Rodney, and others, refused.
5142	The man died at half past twelve that night.
5142	In the course of the forenoon of Tuesday, Murphy (the man since executed) came into the gallery of the sick-house, and was examining the house.

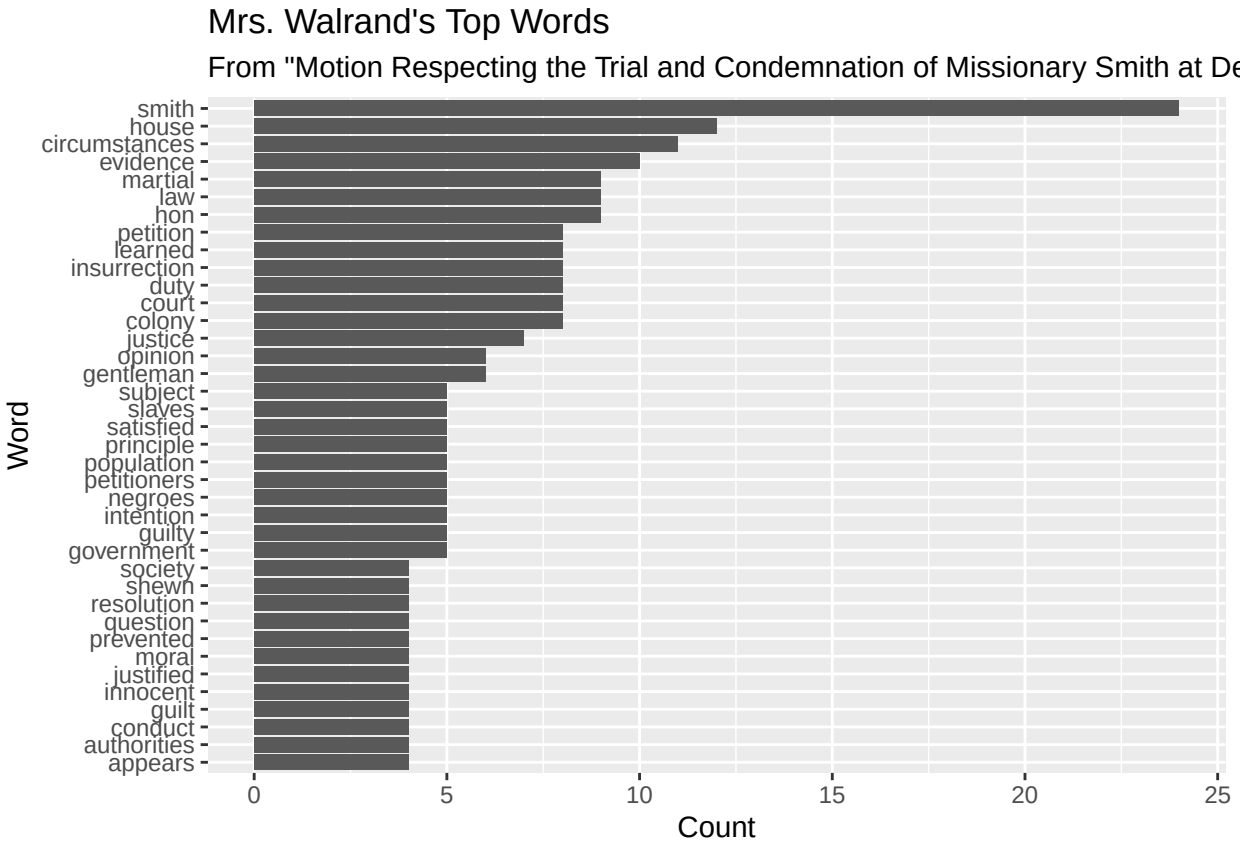
```r
entire_debate <- hansard_1820 %>%
  left_join(file_metadata_1820, hansard_1820, by = "sentence_id") %>%
  filter(debate_id == 5142) %>%
  select("text")

head(entire_debate$text, 20)
```

```
##  [1] "rose, and addressed the House to the following effect:-"
##  [2] "; I confess that, in bringing before this House the question on which I now rise to address you
##  [3] "I cannot conceal from myself, that, even in quarters where one would least have expected it, a
##  [4] "Many persons who have, upon all other occasions, been remarkable for their manly hostility to a
##  [5] "Nay, they would fain fasten upon any excuse to get rid of the subject. \"\""
##  [6] "What signifies inquiring, \"\" say they, \"\"into a transaction which has occurred in a differe
##  [7] "As if distance or climate made any difference in an outrage upon law or justice."
##  [8] "One would have rather expected that the very idea of that distance; the circumstance of the eve
##  [9] "Then, says another, too indolent to inquire, but prompt enough to decide, \"\"It is true there
## [10] "* From the edition published by Hatchard and Son, with the sanction of the London Missionary So
## [11] "ject;"
## [12] "but then every body knows how those petitions are procured, by what descriptions of persons the
## [13] "And, after all, it is merely about a poor missionary!\"\""
## [14] "It is the first time that I have to learn that the weakness of the sufferer; his unprotected si
## [15] "But, it is not enough that he was a Missionary; to make the subject still more unpalatable, for
## [16] "I hasten to this objection, with a view at once to dispose of it."
## [17] "Suppose Mr. Smith had been a methodist; what then?"
## [18] "Does his connexion with that class of religious people, because, on some points essential in th
## [19] "Are British subjects to be treated more or less favourably in courts of law; are they to have a
## [20] "Had he belonged to the society of the methodists, and been employed by the members of that comm
```

```r
mrs_walrand_top_words <- left_join(mrs_walrand_speaker_metadata, hansard_1820) %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words) %>%
  filter(is.na(as.numeric(word))) %>%
  group_by(speaker, word) %>%
  summarize(n = n()) %>%
  arrange(desc(n)) %>%
  top_n(30)

ggplot(data = mrs_walrand_top_words,
       aes(x = reorder(word, n), y = n)) +
  geom_col() +
  coord_flip() +
  labs(title = "Mrs. Walrand\'s Top Words",
       subtitle = "From \"Motion Respecting the Trial and Condemnation of Missionary Smith at Demerara\"
       x = "Word",
       y = "Count")
```



Exercises

1) As we discussed in the beginning of this chapter, analyzing the individual or collective speeches of men in Parliament can give insight into the workings of the Parliament as an institution of power. We can explore the roles of elite white men as they exerted their will–and the will of their peers–through the Parliamentary institution. We can also explore the roles of women in this space and examine how women differ in their contributions, as they were prohibited from becoming members of Parliament. Further, the woman's presence had to be mediated by one of the MPs, as they could not be there without being selected or approved. In this way, women were treated as instruments of power in Parliament.

We can further explore how the role of women manifested in Parliament. First, count the total number of words women spoke in Parliament. To do this, refer to the table below, which visualizes the women in Parliament and the year in which they spoke. Can you find a pattern across the kinds of topics women are brought to Parliament to speak on?

Note: The way in which their names are recorded in this table might not reflect the way in which their names appear in the debates. This is because the names have been consolidated into one representation

Name	Year	Name	Year
Mrs. (Mary Ann) Clark	1809	Mrs. Disraeli	1856
Mrs. Bridgeman	1809	Ms. Cunninghame Graham	1887
Miss Mary Ann Taylor	1809	Mrs. Dillon	1902
Louisa Demont	1820	Mrs. M'Govern	1902
Franchette Martigner	1820	Mrs. Mitchel-Thomson	1907
Mrs. Walrand	1824	Ms. Cave	1907

2) Use the techniques shown above to move between a distant reading of the top words spoken by Mrs. Walrand or Louisa Demont in all of her speeches to the full debate. (As a reminder, you can view all of a speaker's speeches by filtering for just that speaker. You will also need to load the data for the appropriate years.) How does transitioning between the different modes of reading text provide insight into this specific debate? How does this illuminate the role of women in 19th-century Parliament?

3) Adjust the above code to search for a different office title. What can you infer about the different MPs who held this title? Can we support our inferences through close readings of their speeches?

To get you started, here are a few office positions: - Chancellor of the Exchequer - Prime Minister - Attorney General - Lord Chancellor

4) At the beginning of this chapter, we compared counts are based on the "speaker" column with counts based on the "suggested_speaker" column. Expand each table to show 30 rows. How are the two tables different? What are some pros and cons about visualizing the original speaker names and the disambiguated speaker names? Identify a future research question based on what you learn.