8 Working with Full Text

R. Heller · Date: 2026-05-09 · Last changed: 2026-10-31

8.1 Learning objectives

After completing this chapter, you will be able to:

- Extract text from PDF files using `pdftools`
- Apply OCR to scanned documents with `tesseract`
- Perform basic text preprocessing (tokenization, stopword removal)
- Assess when full-text analysis adds value over metadata-only approaches
- Navigate the legal landscape of text and data mining

8.2 Setup

```
library(tidyverse)
library(pdftools)
library(quanteda)
library(glue)

set.seed(20260509)

source(here::here("R", "sci_palette.R"))
```

8.3 Conceptual background

Most bibliometric analyses operate on metadata: titles, abstracts, keywords, and citation links. But metadata captures only a fraction of a paper's content. Full-text analysis opens the door to richer methods: section-level citation context, method extraction, data-availability statements, and fine-grained topic modeling.

The technical pipeline involves three stages: (1) **acquisition** — obtaining PDF files, subject to access rights and TDM (text and data mining) provisions; (2) **extraction** — converting PDF content to machine-readable text; and (3) **preprocessing** — tokenization, normalization, and feature extraction.

`pdftools` wraps the Poppler library to extract text from born-digital PDFs. For scanned documents, `tesseract` provides OCR (optical character recognition). `tabulizer` (based on Tabula) specializes in extracting tables from PDFs. For text preprocessing, `quanteda` and `tm` offer comprehensive toolkits.

Legal frameworks vary by jurisdiction. The EU Copyright Directive (2019/790) provides a TDM exception for research. In the US, TDM is generally considered fair use in research contexts, but publisher terms of service may impose additional restrictions.

8.4 Worked example

8.4.1 Extracting text from a PDF

We demonstrate with a programmatically created PDF to keep this example self-contained.

```
demo_text ← paste(
  "Title: A Review of Bibliometric Methods",
  "",
  "Abstract: This paper reviews current methods in bibliometrics.",
  "We focus on citation analysis, co-authorship networks, and topic modeling.",
  "",
  "1. Introduction",
  "Bibliometrics has grown rapidly since the work of Garfield (1955).",
  "Modern tools enable large-scale analysis of scholarly communication.",
  "",
  "2. Methods",
  "We surveyed 500 papers published between 2020 and 2023.",
  "Data were obtained from OpenAlex.",
  "",
  "3. Conclusions",
  "The field continues to evolve with open data and new computational methods.",
  sep = "\n"
)

tmp_pdf ← tempfile(fileext = ".pdf")
pdf(tmp_pdf, width = 8.5, height = 11)
plot.new()
text(0.05, 0.95, demo_text, adj = c(0, 1), family = "mono", cex = 0.8)
dev.off()
```

```
#> png
#>   2
```

```
extracted ← pdf_text(tmp_pdf)
cat(substr(extracted[1], 1, 500))
```

```
#> Title: A Review of Bibliometric Methods
#>
#> Abstract: This paper reviews current methods in bibliometrics.
#> We focus on citation analysis, co-authorship networks, and topic modeling.
#>
#> 1. Introduction
#> Bibliometrics has grown rapidly since the work of Garfield (1955).
#> Modern tools enable large-scale analysis of scholarly communication.
#>
#> 2. Methods
#> We surveyed 500 papers published between 2020 and 2023.
#> Data were obtained from OpenAlex.
#>
#> 3. Conclusions
#> The field continues to evolve with open data and n
```

8.4.2 Basic text preprocessing

```
corp ← corpus(extracted)
toks ← tokens(corp, remove_punct = TRUE, remove_numbers = TRUE) ▷
  tokens_tolower() ▷
  tokens_remove(stopwords("en"))

dfmat ← dfm(toks)
topfeatures(dfmat, 15)
```

```
#>       methods bibliometrics      analysis             -          data
#>             4             2             2             2             2
#>         title        review  bibliometric      abstract         paper
#>             1             1             1             1             1
#>       reviews       current         focus      citation            co
#>             1             1             1             1             1
```

8.4.3 Word frequency visualization

```
top_words ← topfeatures(dfmat, 15) ▷
  enframe(name = "word", value = "count") ▷
  mutate(word = fct_reorder(word, count))

ggplot(top_words, aes(x = count, y = word)) +
  geom_col(fill = palette_sci(1)) +
  labs(x = "Frequency", y = NULL) +
  theme_sci()
```

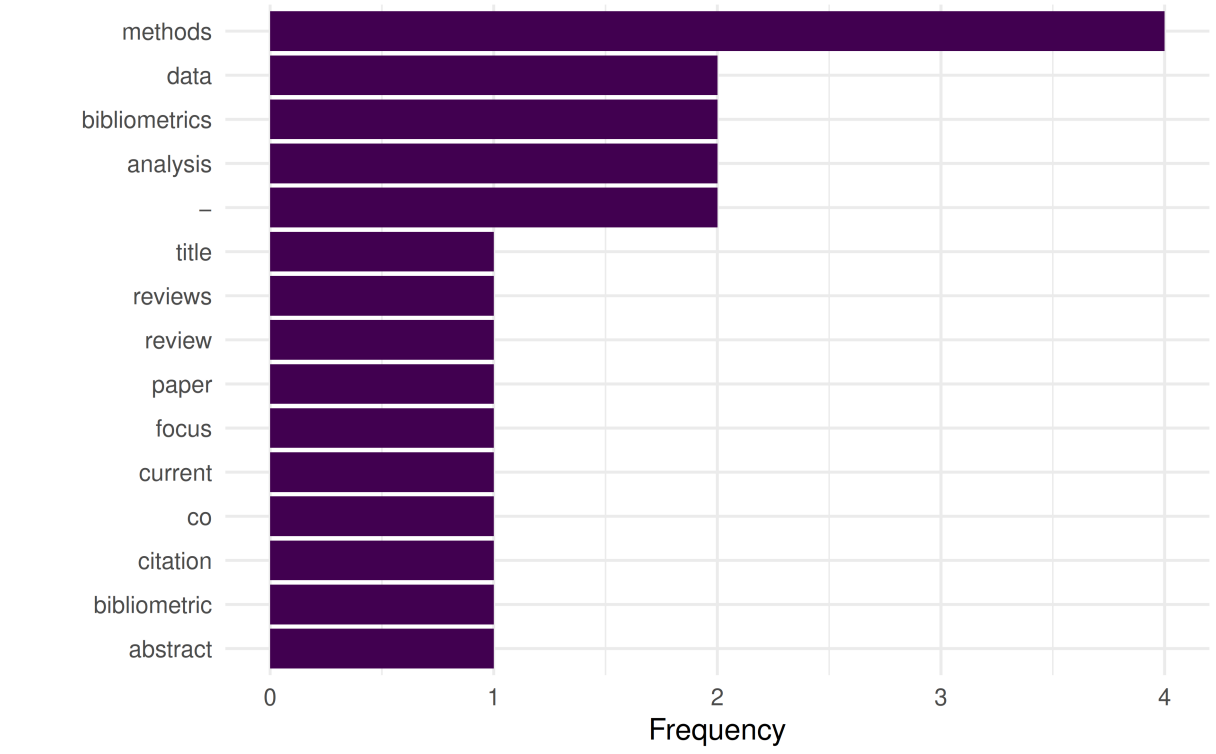


Figure 8.1: Top 15 words extracted from the demo PDF after preprocessing.

8.4.4 OCR with tesseract

```
# Requires tesseract system library and a scanned PDF
# library(tesseract)
# ocr_text ← ocr(scanned_pdf_path, engine = tesseract("eng"))
# cat(substr(ocr_text, 1, 500))
```

8.5 Diagnostics and interpretation

Full-text extraction quality varies dramatically:

- **Born-digital PDFs**: `pdftools` produces clean text, but column layouts may interleave columns.
- **Scanned PDFs**: OCR accuracy depends on scan quality, resolution, and language. Expect 90–98% character accuracy for clean English scans.
- **Figures and equations**: These are lost in text extraction. Do not expect mathematical notation or chart data.
- **Headers and footers**: Page numbers, journal headers, and running titles pollute the extracted text. Consider filtering by position.

8.6 Limitations and responsible use

8.7 Limitations and responsible use

- **Access rights**: Not all PDFs are legally available for TDM. Respect publisher terms and licensing. Open-access articles under CC licenses are generally safe.
- **Extraction errors**: Two-column layouts, ligatures, and non-standard fonts cause parsing artifacts. Always spot-check extracted text.
- **Scalability**: Full-text analysis is orders of magnitude more expensive than metadata analysis. Consider whether the additional information justifies the cost.
- **Bias**: Full text is only available for a subset of publications (typically open access). Analyses based on full text may not generalise to the broader literature (Hicks et al. 2015).

8.8 Common pitfalls

8.9 Common pitfalls

- **Assuming clean extraction.** Even `pdftools` produces artifacts. Always inspect a sample of extracted text before bulk analysis.
- **Skipping preprocessing.** Raw extracted text contains headers, page numbers, and hyphenated line breaks that must be cleaned.
- **Mixing OCR and digital extraction.** Some PDFs contain both digital text and scanned images. Detect which pages need OCR before processing.
- **Ignoring encoding issues.** PDFs may encode characters in non-standard ways. Check for garbled text, especially in references sections.

8.10 Exercises

1. **Extract and compare.** Download an open-access PDF. Extract text with `pdftools`. Compare the extracted abstract with the abstract in OpenAlex. How accurate is the extraction?

2. **Section detection.** Write a function that splits extracted PDF text into sections based on numbered headings ("1. Introduction", "2. Methods", etc.).

3. **Table extraction.** If you have `tabulizer` installed, extract a table from a PDF and convert it to a tibble. What cleaning steps are needed?

8.11 Solutions

Solutions are provided in E.

8.12 Further reading

- Aria and Cuccurullo (2017) — `bibliometrix` supports some full-text features alongside metadata analysis.
- Priem et al. (2022) — OpenAlex provides abstracts for many works, often sufficient without full-text extraction.

8.13 Session info

```
sessionInfo()
```

```
#> R version 4.6.1 (2026-06-24)
#> Platform: x86_64-pc-linux-gnu
#> Running under: Ubuntu 24.04.5 LTS
#>
#> Matrix products: default
#> BLAS:   /usr/lib/x86_64-linux-gnu/openblas-pthread/libblas.so.3
#> LAPACK: /usr/lib/x86_64-linux-gnu/openblas-pthread/libopenblasp-r0.3.26.so;  LAPACK
version 3.12.0
#>
#> locale:
#>  [1] LC_CTYPE=C.UTF-8       LC_NUMERIC=C           LC_TIME=C.UTF-8
#>  [4] LC_COLLATE=C.UTF-8     LC_MONETARY=C.UTF-8    LC_MESSAGES=C.UTF-8
#>  [7] LC_PAPER=C.UTF-8       LC_NAME=C              LC_ADDRESS=C
#> [10] LC_TELEPHONE=C         LC_MEASUREMENT=C.UTF-8 LC_IDENTIFICATION=C
#>
#> time zone: UTC
#> tzcode source: system (glibc)
#>
#> attached base packages:
#> [1] stats     graphics  grDevices utils     datasets  methods   base
#>
#> other attached packages:
#>  [1] quanteda_4.5.1     pdftools_3.9.1     arrow_26.0.1.1     bibliometrix_5.5.0
#>  [5] RefManageR_1.4.0   bib2df_1.1.2.0     rcrossref_1.2.1    gt_1.4.0
#>  [9] tidytext_0.4.3     glue_1.8.1         openalexR_3.1.2    lubridate_1.9.5
#> [13] forcats_1.0.1      stringr_1.6.0      dplyr_1.2.1        purrr_1.2.2
#> [17] readr_2.2.0        tidyr_1.3.2        tibble_3.3.1       ggplot2_4.0.4
#> [21] tidyverse_2.0.0
#>
#> loaded via a namespace (and not attached):
#>   [1] gridExtra_2.3.1        httr2_1.3.0            readxl_1.6.0
#>   [4] rlang_1.3.0            magrittr_2.0.5         otel_0.3.0
#>   [7] compiler_4.6.1         vctrs_0.7.3            httpcode_0.3.0
#>  [10] pkgconfig_2.0.3        fastmap_1.2.0          labeling_0.4.3
#>  [13] ca_0.72                utf8_1.2.6             promises_1.5.0
#>  [16] rmarkdown_2.32         tzdb_0.5.0             bit_4.6.0
#>  [19] xfun_0.62              cachem_1.1.0           jsonlite_2.0.0
#>  [22] SnowballC_0.7.1        later_1.4.8            parallel_4.6.1
#>  [25] stopwords_2.3          R6_2.6.1               bslib_0.12.0
#>  [28] stringi_1.8.9          RColorBrewer_1.1-3     cellranger_1.1.0
#>  [31] jquerylib_0.1.4        assertthat_0.2.1       Rcpp_1.1.2
#>  [34] bookdown_0.48          knitr_1.52             triebeard_0.4.1
#>  [37] base64enc_0.1-6        rentrez_1.2.4          httpuv_1.6.17
#>  [40] Matrix_1.7-5           igraph_2.3.4           timechange_0.4.0
#>  [43] tidyselect_1.2.1       stringdist_0.9.17      pubmedR_1.1.0
#>  [46] rstudioapi_0.19.0      yaml_2.3.12            viridis_0.6.5
#>  [49] miniUI_0.1.2           humaniformat_0.6.0     curl_8.1.0
#>  [52] qpdf_1.4.1             lattice_0.22-9         plyr_1.8.9
#>  [55] shiny_1.15.0           withr_3.0.3            S7_1.0.0
#>  [58] askpass_1.2.1          evaluate_1.0.5         zip_3.0.2
#>  [61] xml2_1.6.0             shinycssloaders_1.1.0  pillar_1.11.1
#>  [64] janeaustenr_1.0.0      DT_0.34.0              plotly_4.12.1
#>  [67] generics_0.1.4         rprojroot_2.1.1        hms_1.1.4
#>  [70] scales_1.4.0           xtable_1.8-8           contentanalysis_1.1.1
#>  [73] tools_4.6.1            data.table_1.18.6.1    tokenizers_0.3.0
#>  [76] openxlsx_4.2.9         XML_3.99-0.25          fs_2.1.0
#>  [79] visNetwork_2.1.4       fastmatch_1.1-8        grid_4.6.1
#>  [82] bibtex_0.5.3           urltools_1.7.3.1       rscopus_0.9.0
#>  [85] dimensionsR_0.0.3      bibliometrixData_0.3.0 cli_3.6.6
#>  [88] viridisLite_0.4.3      downlit_0.4.5          gtable_0.3.6
#>  [91] sass_0.5.0             digest_0.6.39          ggrepel_0.9.8
#>  [94] crul_1.6.0             htmlwidgets_1.6.4      farver_2.1.2
#>  [97] memoise_2.0.1          htmltools_0.5.9        lifecycle_1.0.5
```

```
#> [100] httr_1.4.9             here_1.0.2             mime_0.13
#> [103] bit64_4.8.6
```