22 Topic Modeling

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

22.1 Learning objectives

After completing this chapter, you will be able to:

- Explain the generative assumptions of LDA and STM
- Prepare a document-feature matrix suitable for topic modelling
- Fit LDA and STM models and select the number of topics
- Evaluate topic quality using coherence and exclusivity metrics
- Interpret and label topics using top words and representative documents

22.2 Setup

```
library(tidyverse)
library(openalexR)
library(quanteda)
library(topicmodels)
library(tidytext)
library(stm)
library(glue)
library(gt)

set.seed(20260509)

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

22.3 Conceptual background

A **topic model** is an unsupervised statistical model that discovers latent themes (topics) in a collection of documents. Each topic is a probability distribution over words; each document is a mixture of topics. The model "reverse-engineers" the writing process: given only the observed word frequencies, it infers which topics exist and how they are distributed across documents.

Latent Dirichlet Allocation (LDA) (Blei et al. 2003) is the foundational topic model. It assumes a generative process: for each document, draw a topic mixture from a Dirichlet prior; for each word position, draw a topic from that mixture, then draw a word from the topic's word distribution. Inference (fitting the model) recovers the topic-word and document-topic distributions that best explain the observed data.

The **Structural Topic Model** (STM) (Roberts et al. 2019) extends LDA by allowing document-level covariates (e.g., publication year, journal) to influence topic prevalence and word usage. This is valuable for bibliometrics because we expect topics to vary systematically

with time, discipline, and publication venue. STM also estimates a “content” component that lets word usage within a topic vary by covariate.

Choosing the number of topics (K) is the most consequential modelling decision. Too few topics produce overly broad themes; too many fragment coherent themes into noise. Quantitative approaches include held-out log-likelihood, semantic coherence, and exclusivity metrics. In practice, running models at multiple K values and inspecting the resulting topics is essential.

Topic models have been widely used in scientometrics to map research fields, track thematic evolution, identify emerging areas, and compare the topical focus of journals, institutions, or countries.

22.4 Worked example

22.4.1 Preparing the corpus

```
works ← fetch_openalex(
  entity = "works",
  primary_location.source.id = "S148561398",
  from_publication_date = "2018-01-01",
  to_publication_date = "2023-12-31",
  options = list(sample = 400, seed = 42)
)

text_df ← works ▷
  filter(!is.na(abstract), nchar(abstract) > 100) ▷
  transmute(
    doc_id = id,
    text = paste(display_name, abstract, sep = ". "),
    year = year(publication_date)
  )

cat(glue("Documents for topic modelling: {nrow(text_df)}\n"))
```

```
#> Documents for topic modelling: 98
```

```
corp ← corpus(text_df, docid_field = "doc_id", text_field = "text")
docvars(corp, "year") ← text_df$year

toks ← tokens(corp, remove_punct = TRUE, remove_numbers = TRUE) ▷
  tokens_tolower() ▷
  tokens_remove(stopwords("en")) ▷
  tokens_remove(c("study", "paper", "results", "research", "analysis",
                  "also", "however", "using", "based"))

dfmat ← dfm(toks) ▷
  dfm_trim(min_termfreq = 5, min_docfreq = 3)

cat(glue("DFM: {nrow(dfmat)} docs x {ncol(dfmat)} features\n"))
```

```
#> DFM: 98 docs x 558 features
```

22.4.2 Fitting LDA

```
dtm ← quanteda::convert(dfmat, to = "topicmodels")

lda_model ← LDA(dtm, k = 8, control = list(seed = 42))

lda_topics ← tidy(lda_model, matrix = "beta") ▷
  rename(topic = topic, term = term, beta = beta)

cat(glue("LDA fitted with K = 8 topics\n"))
```

```
#> LDA fitted with K = 8 topics
```

```
lda_top_terms ← lda_topics ▷
  group_by(topic) ▷
  slice_max(beta, n = 10) ▷
  ungroup() ▷
  mutate(term = reorder_within(term, beta, topic))

ggplot(lda_top_terms, aes(x = beta, y = term, fill = factor(topic))) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~ paste("Topic", topic), scales = "free_y", ncol = 4) +
  scale_y_reordered() +
  scale_fill_manual(values = palette_sci(8)) +
  labs(x = "Word probability (beta)", y = NULL) +
  theme_sci(base_size = 9)
```

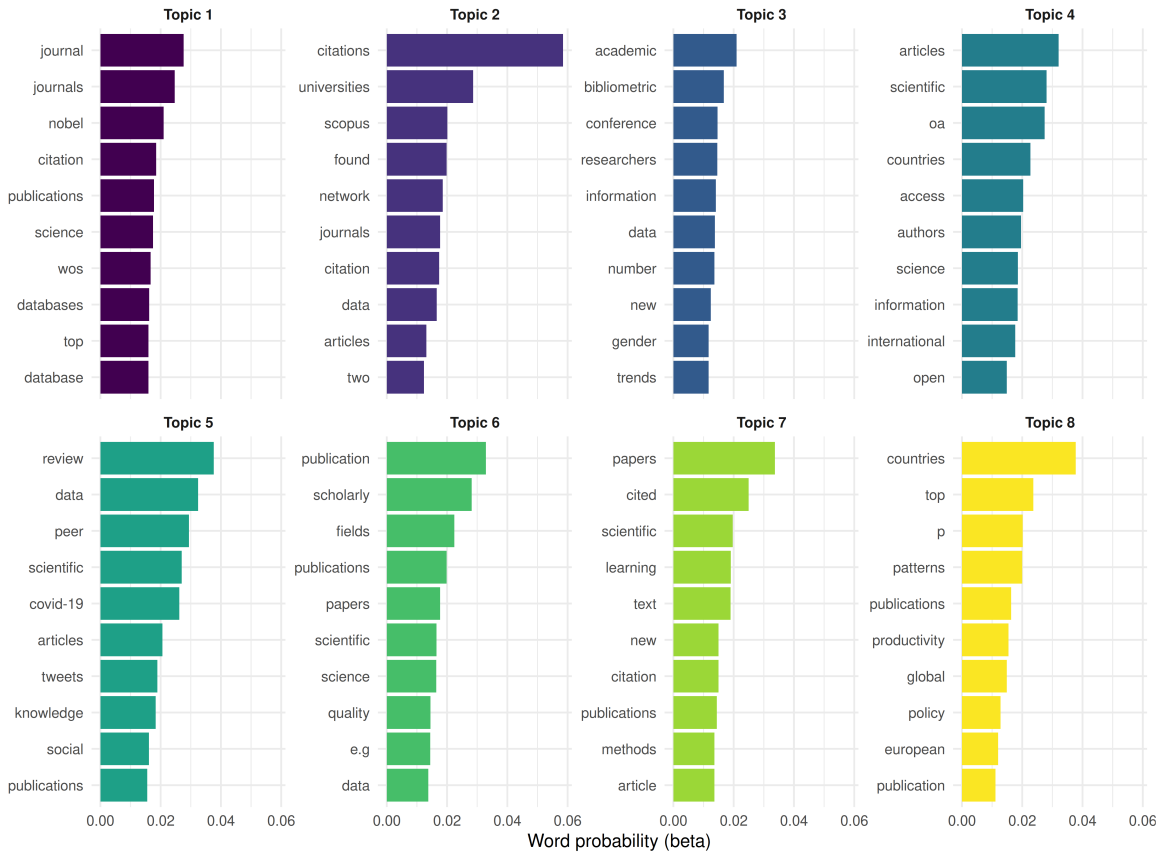


Figure 22.1: Top 10 terms per LDA topic.

22.4.3 Fitting STM with year covariate

```
stm_dfm ← quanteda::convert(dfmat, to = "stm")

stm_model ← stm(
  documents = stm_dfm$documents,
  vocab = stm_dfm$vocab,
  K = 8,
  prevalence = ~ year,
  data = stm_dfm$meta,
  seed = 42,
  verbose = FALSE
)

cat(glue("STM fitted with K = 8 topics, year as prevalence covariate\n"))
```

```
#> STM fitted with K = 8 topics, year as prevalence covariate
```

```
plot(stm_model, type = "summary", n = 5,
     main = "STM Topic Proportions and Top FREX Terms")
```

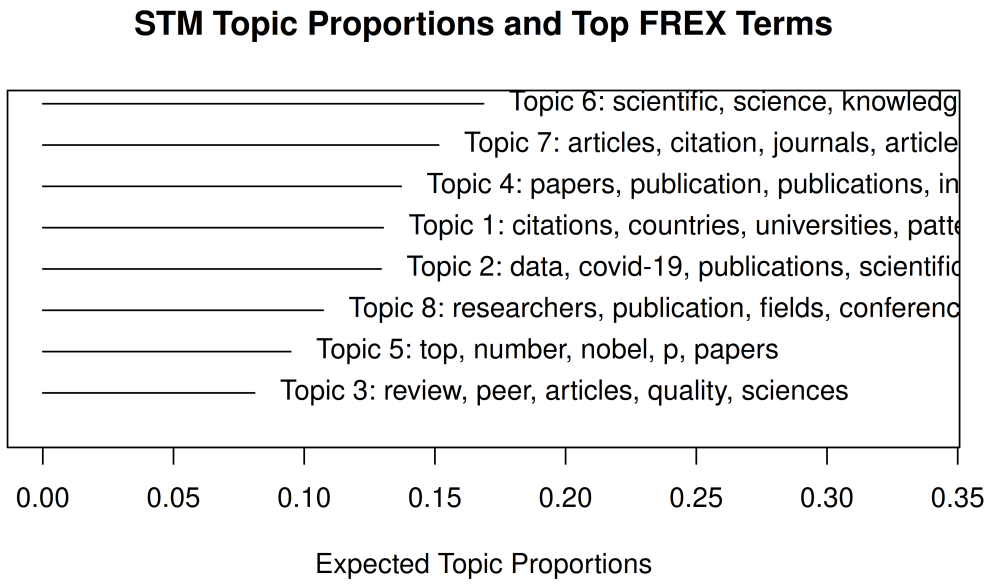


Figure 22.2: Top terms per STM topic, showing FREX (frequent and exclusive) words.

22.4.4 Topic prevalence over time

```r
effect ← estimateEffect(1:8 ~ year, stm_model, meta = stm_dfm$meta)

effect_df ← map_dfr(1:8, function(k) {
  eff ← summary(effect, topics = k)$tables[[1]]
  tibble(
    topic = k,
    term = c("intercept", "year"),
    estimate = eff[, "Estimate"],
    se = eff[, "Std. Error"]
  )
}) ▷
  filter(term == "year")

ggplot(effect_df, aes(x = factor(topic), y = estimate)) +
  geom_col(fill = palette_sci(1)) +
  geom_errorbar(aes(ymin = estimate - 1.96 * se, ymax = estimate + 1.96 * se),
                width = 0.3) +
  geom_hline(yintercept = 0, linetype = "dashed", colour = "grey50") +
  labs(x = "Topic", y = "Year effect on prevalence") +
  theme_sci()
```

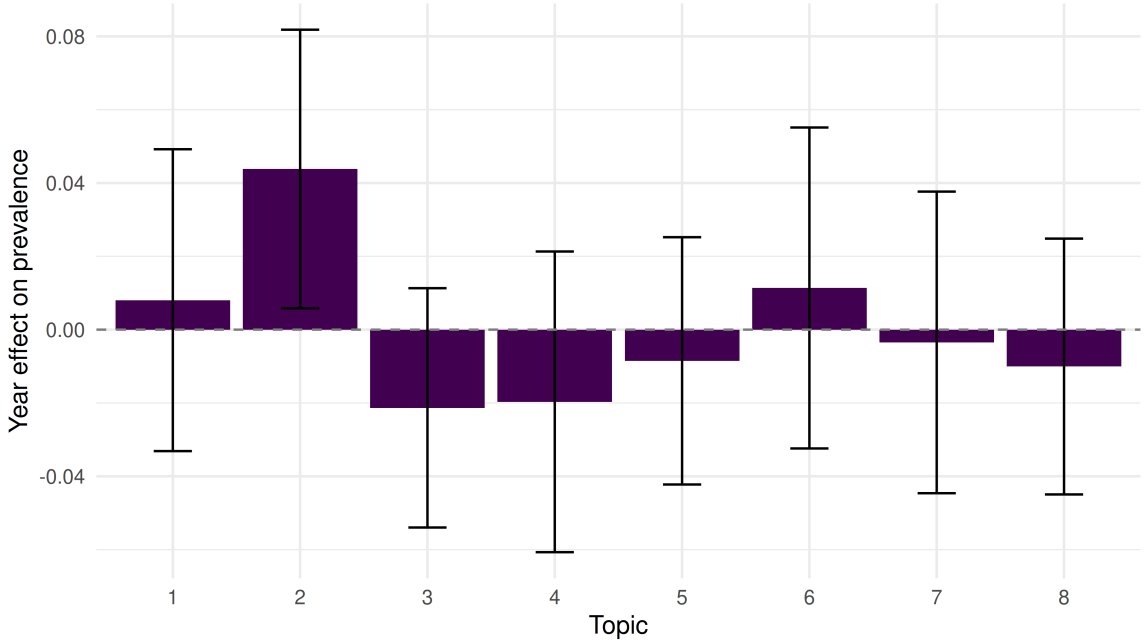


Figure 22.3: Estimated topic prevalence by publication year (STM).

22.4.5 Model diagnostics

```
coherence ← semanticCoherence(stm_model, stm_dfm$documents)
exclusivity ← exclusivity(stm_model)

diag_df ← tibble(
  topic = 1:8,
  coherence = round(coherence, 2),
  exclusivity = round(exclusivity, 2)
)

diag_df ▷ gt()
```

topic	coherence	exclusivity
1	-97.4	9.35
2	-82.0	9.35
3	-94.8	9.65
4	-105.1	9.37
5	-95.6	9.54
6	-77.8	9.01
7	-77.8	9.17
8	-119.7	9.22

22.5 Diagnostics and interpretation

- **Coherence**: High coherence means the top words in a topic tend to co-occur in the same documents. Low coherence indicates a "junk" topic mixing unrelated terms.
- **Exclusivity**: High exclusivity means the top words are distinctive to that topic. Topics with low exclusivity share vocabulary with other topics, suggesting they may need to be split.
- **Coherence-exclusivity trade-off**: Adding more topics generally improves exclusivity but may decrease coherence. Plot both metrics across K values to find a balance.
- **Human validation**: Always read sample documents assigned to each topic. If a topic cannot be labelled by a domain expert, it may not represent a meaningful theme.

22.6 Limitations and responsible use

22.7 Limitations and responsible use

- **Topics are not ground truth.** Topic models are statistical constructs, not natural categories. Different random seeds, preprocessing choices, or K values produce different topics. Never claim a topic model reveals "the" structure of a field.
- **Abstract-only models miss depth.** Abstracts are compressed summaries. Topics derived from abstracts are coarser than those from full text.
- **Number of topics is a researcher choice.** There is no objectively correct K. Sensitivity analysis across multiple K values is essential.
- **Temporal confounds.** In STM, a topic that "grows" over time may reflect genuine intellectual shift or simply increased coverage by the database in recent years (Priem et al. 2022).
- **Replication.** Topic models are sensitive to preprocessing and random initialisation. Report all parameters and fix seeds for reproducibility.

22.8 Common pitfalls

22.9 Common pitfalls

- **Not preprocessing enough.** Rare terms, numbers, and generic academic vocabulary produce noisy topics. Trim the DFM before fitting.
- **Choosing K by perplexity alone.** Held-out perplexity often favours large K values that produce uninterpretable topics. Use coherence and exclusivity alongside perplexity.

- **Treating topic proportions as precise measurements.** Document-topic proportions are probabilistic estimates with uncertainty. Do not report them to three decimal places.
- **Comparing topics across models.** LDA and STM may number and define topics differently. Align topics by content, not by index.

22.10 Exercises

1. **K selection.** Fit LDA models with K = 5, 8, 12, and 15. Compute coherence for each. Plot coherence against K and select the best value. How sensitive are the results?
2. **STM with journal covariate.** Fetch data from two different journals. Fit an STM with journal as a prevalence covariate. Which topics differ most between journals?
3. **Topic labelling.** For the best LDA model, read the titles of the 5 documents with the highest proportion for each topic. Assign a human-readable label to each topic.
4. **LDA vs. STM comparison.** Compare the topics from LDA and STM for the same K. Are they similar? What does the STM's year covariate add?

22.11 Solutions

Solutions are provided in E.

22.12 Further reading

- Blei et al. (2003) — The foundational LDA paper.
- Roberts et al. (2019) — The Structural Topic Model with covariates.
- Silge and Robinson (2017) — *Text Mining with R*: accessible guide to topic modelling in R.
- Aria and Cuccurullo (2017) — `bibliometrix` topic analysis features.

22.13 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] stm_1.3.10                topicmodels_0.2-17
#>  [3] quanteda.textstats_0.97.2 visNetwork_2.1.4
#>  [5] ggraph_2.2.2              tidygraph_1.3.1
#>  [7] igraph_2.3.4              quanteda_4.5.1
#>  [9] pdftools_3.9.1            arrow_26.0.1.1
#> [11] bibliometrix_5.5.0        RefManageR_1.4.0
#> [13] bib2df_1.1.2.0            rcrossref_1.2.1
#> [15] gt_1.4.0                  tidytext_0.4.3
#> [17] glue_1.8.1                openalexR_3.1.2
#> [19] lubridate_1.9.5           forcats_1.0.1
#> [21] stringr_1.6.0             dplyr_1.2.1
#> [23] purrr_1.2.2               readr_2.2.0
#> [25] tidyr_1.3.2               tibble_3.3.1
#> [27] ggplot2_4.0.4             tidyverse_2.0.0
#>
#> loaded via a namespace (and not attached):
#>   [1] bibtex_0.5.3           RColorBrewer_1.1-3     rstudioapi_0.19.0
#>   [4] jsonlite_2.0.0         magrittr_2.0.5         modeltools_0.2-25
#>   [7] farver_2.1.2           rmarkdown_2.32         fs_2.1.0
#>  [10] vctrs_0.7.3            memoise_2.0.1          askpass_1.2.1
#>  [13] base64enc_0.1-6        htmltools_0.5.9        contentanalysis_1.1.1
#>  [16] curl_8.1.0             janeaustenr_1.0.0      cellranger_1.1.0
#>  [19] sass_0.5.0             bslib_0.12.0           htmlwidgets_1.6.4
#>  [22] tokenizers_0.3.0       plyr_1.8.9             httr2_1.3.0
#>  [25] plotly_4.12.1          cachem_1.1.0           dimensionsR_0.0.3
#>  [28] mime_0.13              lifecycle_1.0.5        pkgconfig_2.0.3
#>  [31] Matrix_1.7-5           R6_2.6.1               fastmap_1.2.0
#>  [34] shiny_1.15.0           digest_0.6.39          patchwork_1.3.2
#>  [37] shinycssloaders_1.1.0  rprojroot_2.1.1        SnowballC_0.7.1
#>  [40] labeling_0.4.3         urltools_1.7.3.1       timechange_0.4.0
#>  [43] polyclip_1.10-7        httr_1.4.9             compiler_4.6.1
#>  [46] here_1.0.2             bit64_4.8.6            withr_3.0.3
#>  [49] S7_1.0.0               viridis_0.6.5          ggforce_0.5.0
#>  [52] MASS_7.3-65            bibliometrixData_0.3.0 tools_4.6.1
#>  [55] otel_0.3.0             stopwords_2.3          zip_3.0.2
#>  [58] httpuv_1.6.17          rentrez_1.2.4          promises_1.5.0
#>  [61] grid_4.6.1             stringdist_0.9.17      reshape2_1.4.5
#>  [64] generics_0.1.4         gtable_0.3.6           tzdb_0.5.0
#>  [67] rscopus_0.9.0          ca_0.72                data.table_1.18.6.1
#>  [70] hms_1.1.4              xml2_1.6.0             utf8_1.2.6
#>  [73] ggrepel_0.9.8          pillar_1.11.1          nsyllable_1.0.1
```

```
#>  [76] vroom_1.7.1            later_1.4.8            tweenr_2.0.3
#>  [79] lattice_0.22-9         bit_4.6.0              tidyselect_1.2.1
#>  [82] tm_0.7-20              miniUI_0.1.2           downlit_0.4.5
#>  [85] knitr_1.52             gridExtra_2.3.1        NLP_0.3-3
#>  [88] bookdown_0.48          stats4_4.6.1           crul_1.6.0
#>  [91] xfun_0.62              graphlayouts_1.2.5     matrixStats_1.5.0
#>  [94] DT_0.34.0              humaniformat_0.6.0     stringi_1.8.9
#>  [97] qpdf_1.4.1             yaml_2.3.12            evaluate_1.0.5
#> [100] httpcode_0.3.0         cli_3.6.6              xtable_1.8-8
#> [103] jquerylib_0.1.4        Rcpp_1.1.2             readxl_1.6.0
#> [106] triebeard_0.4.1        XML_3.99-0.25          parallel_4.6.1
#> [109] assertthat_0.2.1       pubmedR_1.1.0          slam_0.1-56
#> [112] viridisLite_0.4.3      scales_1.4.0           crayon_1.5.3
#> [115] openxlsx_4.2.9         rlang_1.3.0            fastmatch_1.1-8
```