CSCI E-89B Introduction to Natural Language Processing

Harvard Extension School

Dmitry Kurochkin

Fall 2025
Lecture 8

Contents

Contents

Recap of Latent Dirichlet Allocation (LDA)

- **Generative Process:**
  - $\theta_m \sim$ Dirichlet(α): Topic distribution for each document m of length N.
  - For each word $w_n (n = 1, 2, \ldots, N)$ in document m:
    $z_n \sim$ Multinomial(θ_m) : Select a topic.
    $w_n \sim$ Multinomial(β_{z_n}) : Generate a word from the topic.
- **Parameters:**
  - α: Controls the document-topic distribution sparsity.
  - β: Controls the topic-word distribution.

Reference: Blei DM, Ng A, Jordan M (2003). "Latent Dirichlet Allocation." Journal of Machine Learning Research, 3, 993–1022.

Limitations of LDA

- **No Covariate Modelling:**
  - LDA does not incorporate document-level metadata such as:
    - ★ Author identification
    - ★ Publication date
    - ★ Source or publication name
    - ★ Geographic location
    - ★ Topic-specific tags or keywords
  - Lacks ability to model the impact of these factors on topic distributions.
- **Consequences:**
  - Potential loss of contextual relevance in topic assignments.
  - Reduced model flexibility and applicability to real-world data with metadata.
- **Advancements:**
  - Structural Topic Modeling (STM) and other models incorporate these factors for better analysis and insights.

Contents

STM Mathematical Formulation

- **Objective:** Extend topic models by incorporating document-level metadata.
- **Generative Process:**
  - **Topic Prevalence:**
    - ★ Draw document-topic attention from a logistic-normal model:

$$\theta_d | X_d\gamma, \Sigma \sim \text{LogisticNormal}(\mu = X_d\gamma, \Sigma)$$

    - ★ Where X_d is a 1-by-p vector (document covariates), γ is a p-by-$(K-1)$ matrix of coefficients, and Σ is a $(K-1)$-by-$(K-1)$ covariance matrix.
  - **Topic Content:**
    - ★ Form the word distribution for each topic k:

$$\beta_{d,k} \propto \exp(m + \kappa_k^{(t)} + \kappa_{y_d}^{(c)} + \kappa_{y_d,k}^{(i)})$$

    - ★ Where m and each $\kappa_k^{(t)}$, $\kappa_{y_d}^{(c)}$, and $\kappa_{y_d,k}^{(i)}$ are V-length (size of vocabulary) vectors representing words in the vocabulary.
  - **Word Generation:**
    - ★ For each word n in document d:

$$z_{n,d} | \theta_d \sim \text{Multinomial}(\theta_d) : \text{Topic assignment.}$$

$$w_{n,d} | z_{n,d}, \beta_{d,k} \sim \text{Multinomial}(\beta_{d,k=z_{n,d}}) : \text{Generate a word.}$$

Contents

Advantages of STM

- **Metadata Integration:**
  - STM incorporates document-level metadata, such as author, publication date, and category, allowing the capture of contextual influences on topic prevalence.
  - Adjusts topic-word distributions based on metadata, improving the model's ability to reflect real-world linguistic and contextual nuances.
- **Advanced Analytical Capabilities:**
  - Allows for hypothesis testing regarding the effect of covariates on topics, providing a framework for causal analysis.
  - Facilitates detailed exploration of how different document attributes impact thematic content, supporting nuanced insights.
- **Enhanced Flexibility:**
  - STM's adaptability makes it applicable across various fields such as social sciences, digital humanities, and market research.
  - Supports the modeling of diverse text types, from academic articles to social media content, enabling broad applicability.

Advantages of STM (Continued)

- **Improved Topic Interpretability:**
  - Yields more interpretable topic models by accounting for external influences, which can clarify topic definitions and relationships.
  - Helps identify how different contexts modify thematic expressions, enhancing the model's explanatory power.
- **Robust Statistical Framework:**
  - Utilizes variational inference techniques, effectively handling large datasets and providing scalable solutions.
  - Offers comprehensive diagnostics and validation tools for model evaluation, ensuring accurate and reliable analyses.

Contents

Applications of STM

- **Media Framing Analysis:**
  - Explore how different media outlets present issues, highlighting shifts in narratives over time.
  - Analyze the influence of media bias on public perception by decomposing coverage into distinct topics.
- **Survey Responses:**
  - Extract and quantify themes in open-ended survey questions, allowing for large-scale qualitative analyses.
  - Identify correlations between demographic data and thematic concerns, uncovering latent trends in respondent attitudes.
- **Social Media:**
  - Monitor emerging trends and track public sentiment on social platforms in real-time.
  - Analyze how topics evolve in response to current events, providing insights into consumer and public reactions.

Applications of STM (Continued)

- **Academic Research:**
  - Facilitate meta-analysis across vast collections of academic papers by discerning shifts in research focus and emerging fields.
  - Enable content analysis of large corpuses, such as historical records or literary texts, highlighting thematic developments.
- **Business Intelligence:**
  - Derive competitive insights by analyzing customer feedback and reviews to identify product strengths and weaknesses.
  - Use thematic analysis to steer strategic decisions in marketing, identifying key brand perceptions and consumer priorities.

Introduction to the 'stm' Package

- **Installation:**
  - Install the package directly from CRAN using `install.packages("stm")`, ensuring integration with your existing R workflow.
  - Compatible with various R environments, allowing flexibility across different systems.
- **Key Features:**
  - **Pre-processing Functions:**
    - ★ Tailored functions for cleaning and preparing text data, including tokenization, stopword removal, and stemming, to optimize text for modeling.
    - ★ Supports metadata inclusion for context-aware processing and analysis.
  - **Flexible Model Fitting:**
    - ★ Accommodates a wide range of covariates, allowing for complex analytic structures in modeling topics.
    - ★ Implements advanced statistical techniques like variational inference for efficient model estimation and convergence.

Advanced Visualization and Support

- **Visualization Tools:**
  - Interactive visual outputs facilitate deep dives into topic structures and thematic mappings, enhancing interpretative clarity.
  - Plotting capabilities to explore metadata, visualize topic prevalence, and assess covariate effects in a user-friendly manner.
- **Support and Resources:**
  - **Comprehensive Documentation:**
    - ★ Extensive online resources, including tutorials and case studies, guide users from basic setups to advanced analyses.
  - **Active Community and Development:**
    - ★ Regular updates and community-driven support on platforms like GitHub, promoting collaborative improvements and new feature requests.
    - ★ Access to a vibrant user community that shares insights and solutions to common challenges, fostering a collaborative learning environment.

R Example

```r
# Complete set of sample documents
documents <- c(
  # Author1 - Cats Focus
  "Cats purr gently and climb high trees. Chasing a mouse is fun for cats.",
  "Independent creatures, cats enjoy solitude and love their nap time.",
  "A purring cat climbed the bookshelf, watching over the room.",
  "Cats meow softly and purr when content, loving to stretch in the sun.",
  "Cats whisk their tails as they leap onto furniture, quietly observing.",
  "Kittens sleep all day, curled up in warmth, dreaming peacefully.",
  "Cats bat at light reflections, intrigued by the moving spots.",
  "Cats explore shaded gardens, watching insects dart across leaves.",

  # Author2 - Dogs Focus
  "Dogs bark loudly at strangers and fetch sticks with enthusiasm.",
  "Loyal dogs accompany humans on hikes and love to chase balls.",
  "When the energetic dog spotted a squirrel, it barked energetically.",
  "Regularly, dogs enjoy long walks, sniffing and exploring their environment.",
  "Dogs dig in the garden, uncovering hidden treasures with muddy paws.",
  "Playful dogs bound through play areas, chasing each other joyously.",
  "Energetic dogs swim in the lake, splashing happily in the cool water.",
  "Dogs lounge on porches, enjoying the sunshine with a content sigh."
)

# Corresponding metadata with only the author covariate
metadata <- data.frame(
  author = rep(c("Author1", "Author2"), each = 8)
)
```

R Example (Continued)

```r
# Load necessary library
library(stm)

# Set a seed for reproducibility
set.seed(54321)

# Preprocess text data
processed <- textProcessor(documents = documents, metadata = metadata, lowercase = TRUE)
out <- prepDocuments(processed$documents, processed$vocab, processed$meta)

# Assign preprocessed variables
docs <- out$documents
vocab <- out$vocab
meta <- out$meta

# Fit the STM model with the expanded document set
stm_model <- stm(documents = docs, vocab = vocab, K = 2,
                 prevalence = ~ author,  # Analyzing the influence of author
                 data = meta,
                 max.em.its = 100,
                 init.type = "Spectral")
```

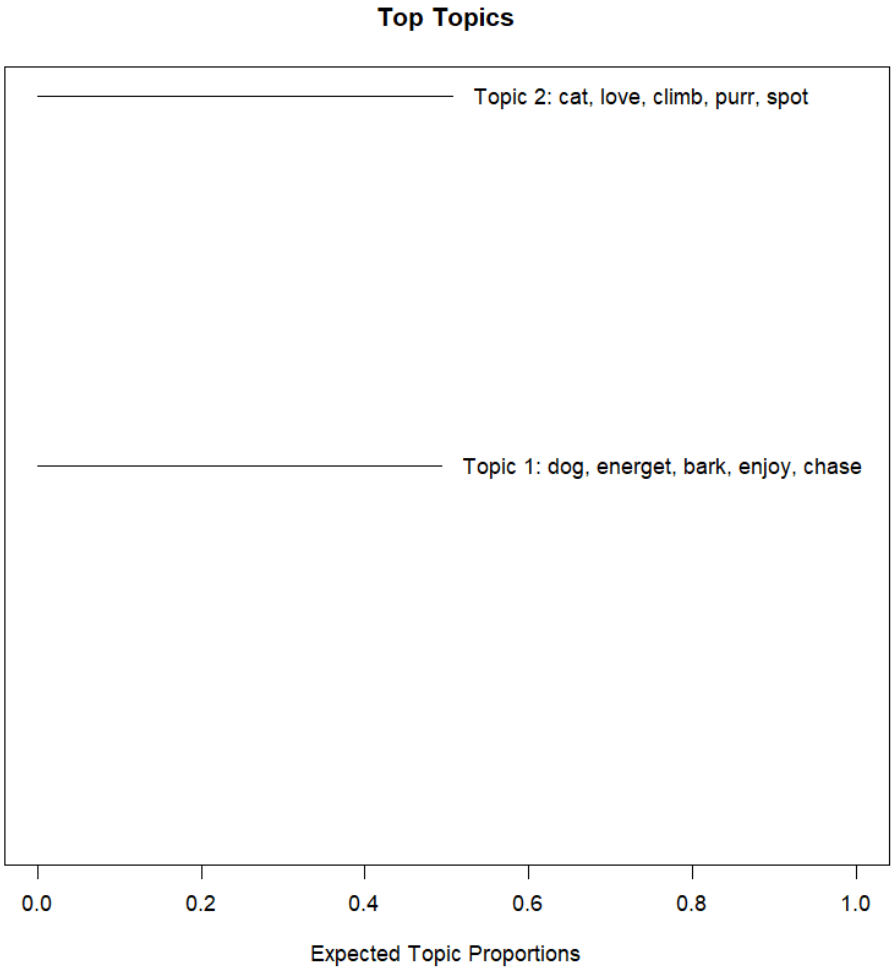
R Example (Continued)

```r
# Summarize the fitted model
summary(stm_model)
```

```
A topic model with 2 topics, 15 documents and a 14 word dictionary.
Topic 1 Top Words:
   Highest Prob: dog, energet, bark, enjoy, chase, explor, content
   FREX: dog, energet, bark, enjoy, chase, explor, content
   Lift: energet, dog, bark, enjoy, chase, explor, content
   Score: bark, dog, energet, enjoy, chase, spot, cat
Topic 2 Top Words:
   Highest Prob: cat, love, climb, purr, spot, watch, garden
   FREX: cat, climb, purr, spot, watch, love, garden
   Lift: cat, climb, purr, spot, watch, love, garden
   Score: cat, spot, climb, watch, purr, love, garden
```

R Example (Continued)

```
# Visualize topics
plot(stm_model, type = "summary", n = 5)
```



R Example (Continued)

```r
# Estimate effects of author on topics
effects <- estimateEffect(1:2 ~ author, stmobj = stm_model,
                          metadata = meta, uncertainty = "Global")

# Plot the effects with confidence intervals
plot(effects, covariate = "author", topics = 1:2, model = stm_model,
     method = "difference", cov.value1 = "Author1", cov.value2 = "Author2",
     xlab = "More Author 1 ... More Author 2",
     main = "Effect of Author Across Topics",
     labeltype = "custom", custom.labels = paste("Topic", 1:2))
```