Computational Sociology

Topic Modeling

Dr. Thomas Davidson

Rutgers University

March 21, 2024

Plan

1. Course updates
2. Introduction to topic modeling
3. Latent Dirichlet Allocation (LDA)
4. Structural Topic Modeling (STM)

Course updates

Feedback on proposals and homework 2

- Comments on proposals sent via Slack
- Homework 2 scores and comments available on Github

Course updates

Homework 3

- Homework 3 on NLP will be released by the end of the week.
  - Due date is 3/29 at 5pm
  - Topics:
    - Intro to NLP
    - Word embedding
    - Topic modeling

Course updates

Progress on projects

- Initial data collection due 4/8 at 5pm.*
  - Preliminary analysis of some or all of the data you will use in your project; consider this as a draft of the Data section of your manuscript
    - Description of the data collection process
    - Preliminary data analysis
    - Include 1-2 summary tables and 1-2 visualizations

* Syllabus states 4/5 but extending deadline by a few days.

Course updates

Progress on projects

- Submit the following via email (include SOC577 in subject line):
    - A link to a Github repository (add me as a collaborator if it is private)
        - Code used to collect and analyze data (ideally in R/RMarkdown)
        - A document containing the analysis (ideally rendered using RMarkdown, but a PDF of a Word document is fine)

Introduction to topic modeling

What is topic modeling?

- A corpus of documents contains a set of latent themes or "topics"
- A topic model is a probabilistic algorithm is designed to inductively create a "model" of these latent topics
- This gives us an overview of the content of an entire corpus and the ability to characterize individual documents

Introduction to topic modeling

Topic modeling and sociology (Mohr and Bogdanov 2013)

- Topic modeling is a "lens" to allow reading of a corpus "in a different light and at a different scale" to traditional content analsyis.
  - Often we want to conduct an automated coding of a large corpus, but it can also be helpful for a "close reading" of a smaller corpus

Introduction to topic modeling

Topic modeling and sociology (Mohr and Bogdanov 2013)

- The approach still requires interpretation, but produces "a fundamental shift in the locus of methodological subjectivity", "from pre-counting to post-counting" (561).
  - Traditional content analysis requires us to develop our categories of analysis before coding a corpus, whereas the interpretative phase of LDA occurs after we have trained a model
  - "topic models do not remove the scholarly or the hermeneutic work from the project of analyzing a textual corpus, topic models simply move the bulk of this labor over to the other side of the data modeling procedure."

Introduction to topic modeling

Topic modeling and sociology (DiMaggio, Nag, and Blei 2013)

- Topic modeling is “an inductive relational approach to the study of culture”
  - It is *inductive* because the topics are “discovered” from the text
  - It is *relational* because meaning emerges out of relationships between words
- Topic modeling provides a high level of “substantive interpretability”
- The approach can discover different theoretical properties of language of interest to sociologists of culture
  - Framing
  - Polysemy
  - Heteroglossia

Introduction to topic modeling

Topic modeling and sociology (DiMaggio, Nag, and Blei 2013)

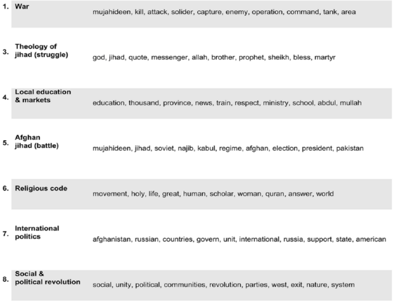
"Topic modeling will not be a panacea for sociologists of culture. But it is a powerful tool for helping us understand and explore large archives of texts. Used properly by subject-area experts with appropriate validation, topic models can be valuable complements to other interpretive approaches, offering new ways to operationalize key concepts and to make sense of large textual corpora."

Introduction to topic modeling

Topic modeling and sociology (Karell and Freedman 2019)

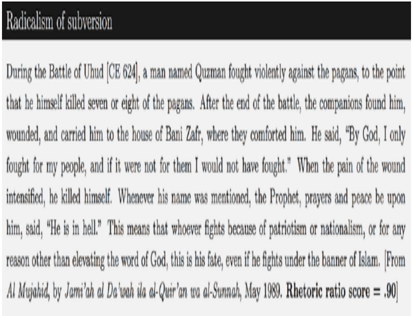
- Karell and Freedman use topic modeling to study the rhetoric of militant groups in Afghanistan
  - They characterize the process as *computational abduction* and describe "a recursive movement between the computational results, a close reading of selected corpus material, further literature on social movements, and additional theories that engendered a novel conceptualization of radical rhetorics."
  - The topic model results are used to characterize the discourse of different militant groups.

Rhetorics of radicalism



1.	War	mujahideen, kill, attack, solider, capture, enemy, operation, command, tank, area
3.	Theology of jihad (struggle)	god, jihad, quote, messenger, allah, brother, prophet, sheikh, bless, martyr
4.	Local education & markets	education, thousand, province, news, train, respect, ministry, school, abdul, mullah
5.	Afghan jihad (battle)	mujahideen, jihad, soviet, najib, kabul, regime, afghan, election, president, pakistan
6.	Religious code	movement, holy, life, great, human, scholar, woman, quran, answer, world
7.	International politics	afghanistan, russian, countries, govern, unit, international, russia, support, state, american
8.	Social & political revolution	social, unity, political, communities, revolution, parties, west, exit, nature, system

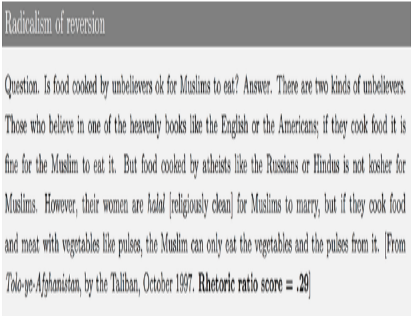
Rhetorics of radicalism



Radicalism of subversion

During the Battle of Uhud [CE 624], a man named Quzman fought violently against the pagans, to the point that he himself killed seven or eight of the pagans. After the end of the battle, the companions found him, wounded, and carried him to the house of Bani Zafr, where they comforted him. He said, "By God, I only fought for my people, and if it were not for them I would not have fought." When the pain of the wound intensified, he killed himself. Whenever his name was mentioned, the Prophet, prayers and peace be upon him, said, "He is in hell." This means that whoever fights because of patriotism or nationalism, or for any reason other than elevating the word of God, this is his fate, even if he fights under the banner of Islam. [From *Al Mujahid*, by *Jami'ah al Da'wah ila al-Quir'an wa al-Sunnah*, May 1989. **Rhetoric ratio score = .90**]

Rhetorics of radicalism



Radicalism of reversion

Question. Is food cooked by unbelievers ok for Muslims to eat? Answer. There are two kinds of unbelievers. Those who believe in one of the heavenly books like the English or the Americans; if they cook food it is fine for the Muslim to eat it. But food cooked by atheists like the Russians or Hindus is not kosher for Muslims. However, their women are *halal* [religiously clean] for Muslims to marry, but if they cook food and meat with vegetables like pulses, the Muslim can only eat the vegetables and the pulses from it. [From *Tolo-ye-Afghanistan*, by the Taliban, October 1997. **Rhetoric ratio score = .29**]

Rhetorics of radicalism

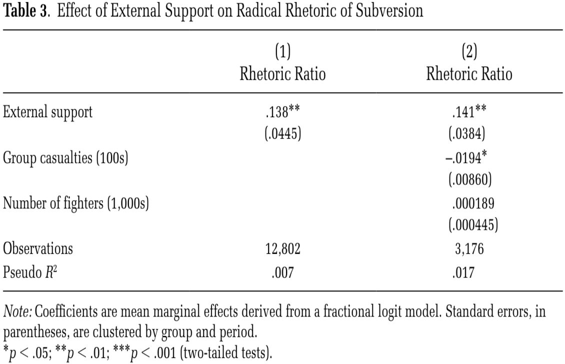


Table 3. Effect of External Support on Radical Rhetoric of Subversion

	(1) Rhetoric Ratio	(2) Rhetoric Ratio
External support	.138**	.141**
	(.0445)	(.0384)
Group casualties (100s)		–.0194*
		(.00860)
Number of fighters (1,000s)		.000189
		(.000445)
Observations	12,802	3,176
Pseudo R^2	.007	.017

Note: Coefficients are mean marginal effects derived from a fractional logit model. Standard errors, in parentheses, are clustered by group and period.

$^{*}p < .05$; $^{**}p < .01$; $^{***}p < .001$ (two-tailed tests).

Introduction to topic modeling

Social movement framing (Davidson 2024)

- In a recent *Mobilization* paper, I use topic modeling to analyze discourse of Britain First (BF), a far-right organization in the UK
- Topic model trained using Facebook posts by BF
  - Topics grouped into broader metatopics
- Results in regression analysis to measure framing

Social movement framing

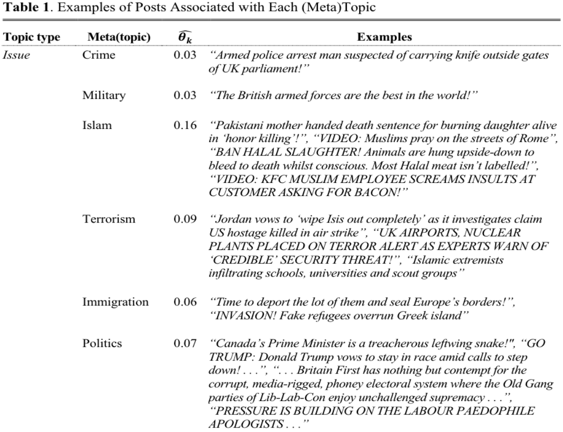
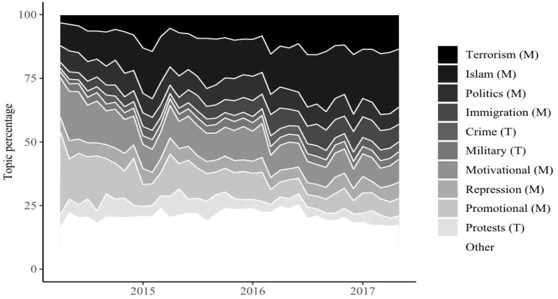


Table 1. Examples of Posts Associated with Each (Meta)Topic

Topic type	Meta(topic)	$\widehat{\theta_k}$	Examples
Issue	Crime	0.03	*"Armed police arrest man suspected of carrying knife outside gates of UK parliament!"*
	Military	0.03	*"The British armed forces are the best in the world!"*
	Islam	0.16	*"Pakistani mother handed death sentence for burning daughter alive in 'honor killing'!", "VIDEO: Muslims pray on the streets of Rome", "BAN HALAL SLAUGHTER! Animals are hung upside-down to bleed to death whilst conscious. Most Halal meat isn't labelled!", "VIDEO: KFC MUSLIM EMPLOYEE SCREAMS INSULTS AT CUSTOMER ASKING FOR BACON!"*
	Terrorism	0.09	*"Jordan vows to 'wipe Isis out completely' as it investigates claim US hostage killed in air strike", "UK AIRPORTS, NUCLEAR PLANTS PLACED ON TERROR ALERT AS EXPERTS WARN OF 'CREDIBLE' SECURITY THREAT!", "Islamic extremists infiltrating schools, universities and scout groups"*
	Immigration	0.06	*"Time to deport the lot of them and seal Europe's borders!", "INVASION! Fake refugees overrun Greek island"*
	Politics	0.07	*"Canada's Prime Minister is a treacherous leftwing snake!", "GO TRUMP: Donald Trump vows to stay in race amid calls to step down! . . .", ". . . Britain First has nothing but contempt for the corrupt, media-rigged, phoney electoral system where the Old Gang parties of Lib-Lab-Con enjoy unchallenged supremacy . . .", "PRESSURE IS BUILDING ON THE LABOUR PAEDOPHILE APOLOGISTS . . ."*

Social movement framing

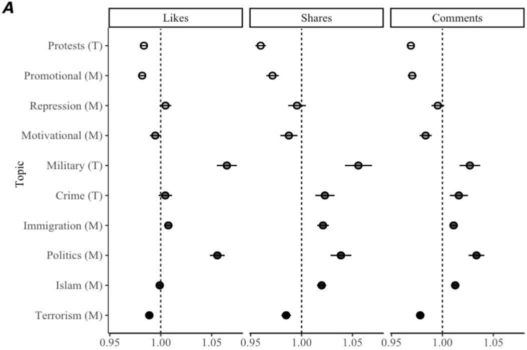
Figure 3. Topic Prevalence Over Time



Note: (Meta)topic prevalence over time, four-week averages. Topic names are prefixed to denote whether each is an individual topic ("T") or a metatopic ("M"). The Other category, shaded in white, represents residual topics not considered in the analysis.

Social movement framing

Figure 6. Topics, Engagement Bait, and Engagement Metrics



Latent Dirichlet Allocation

Naming

- We observe the documents but the topics are **latent**
- The model is based on a probability distribution called the **Dirichlet** distribution
- Using this model we **allocate** words to topics

Latent Dirichlet Allocation

Intuition

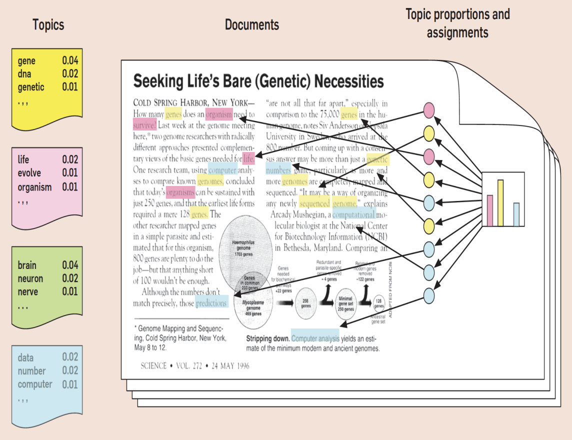
- A **topic** is a distribution over a vocabulary
    - Each word in the vocabulary has a probability of belonging to the topic
- Let's say we train an LDA on a newspaper corpus
    - We find a topic that seems to capture information about *sports*
        - The words "football" and "goal" have a high probability
        - The words "literary" and "helicopter" have a low probability
    - This topic can be represented as a distribution over all words in the vocabulary
        - $Topic\ _k = [football : 0.34, goal : 0.23, ..., literary : 0.0001, helicopter : 0.0002, ...]$

Latent Dirichlet Allocation

Intuition

- A **document** is a distribution over topics
    - *All* documents contain *all* topics, but in different proportions
- Let's say we take an article about a new player a football team hired and look at the topics based on the newspaper model:
    - The highest probability topic might be *Sports*, but the article also disusses contract and the position of the player in the labor market
        - Thus the document may also contain the topics *Finance* and *Labor*.
    - The article is irrelevant to other issues in discussed in newspapers, so has a low probability of containing the topic *national security* or *arts*.
    - $Document_d = [Sports : 0.63, Finance : 0.25, Labor : 0.12, ..., National\ security : 0.001, Arts : 0.002, ...]$

Latent Dirichlet Allocation


Topics
Documents
Topic proportions and assignments
gene 0.04
dna 0.02
genetic 0.01
,,,
life 0.02
evolve 0.01
organism 0.01
,,,
brain 0.04
neuron 0.02
nerve 0.01
...
data 0.02
number 0.02
computer 0.01
,,,
Haemophilus genome 1703 genes
Genes in common 233 genes
Mycoplasma genome 469 genes
Genes needed for biochemical pathways +22 genes
256 genes
Redundant and parasite-specific genes removed − 4 genes
Minimal gene set 250 genes
Related and modern genes removed −122 genes
128 genes
Ancestral gene set
ADAPTED FROM NCBI

Seeking Life's Bare (Genetic) Necessities

COLD SPRING HARBOR, NEW YORK— How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

Although the numbers don't match precisely, those predictions "are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an

Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

SCIENCE • VOL. 272 • 24 MAY 1996

Blei, David M. 2012. "Probabilistic Topic Models." Communications of the ACM 55 (4): 77. https://doi.org/10.1145/2133806.2133826.

Latent Dirichlet Allocation

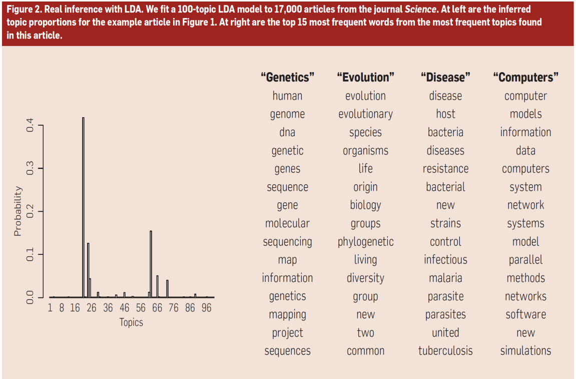


Figure 2. Real inference with LDA. We fit a 100-topic LDA model to 17,000 articles from the journal *Science*. At left are the inferred topic proportions for the example article in Figure 1. At right are the top 15 most frequent words from the most frequent topics found in this article.

"Genetics"	"Evolution"	"Disease"	"Computers"
human	evolution	disease	computer
genome	evolutionary	host	models
dna	species	bacteria	information
genetic	organisms	diseases	data
genes	life	resistance	computers
sequence	origin	bacterial	system
gene	biology	new	network
molecular	groups	strains	systems
sequencing	phylogenetic	control	model
map	living	infectious	parallel
information	diversity	malaria	methods
genetics	group	parasite	networks
mapping	new	parasites	software
project	two	united	new
sequences	common	tuberculosis	simulations

Blei, David M. 2012. "Probabilistic Topic Models." Communications of the ACM 55 (4): 77. https://doi.org/10.1145/2133806.2133826.

Latent Dirichlet Allocation

Intuition

- Topic modeling is a *generative* process*
    - The goal is to create a plausible model that can mimic the hidden structure and *generate* the observed documents
    - ""The utility of topic models stems from the property that the inferred hidden structure resembles the thematic structure of the collection." Blei, 2012.

* Note this is term is also used to refer to generative AI, but unlike ChatGPT and other large language models we cannot use topic models to generate plausible texts.

Latent Dirichlet Allocation

Mathematical formulation

- There are four components that we need to compute the LDA over a corpus of documents:
  1. Observed words $w_{1:D}$, where $w_{d,n}$ is the n^{th} word in document d.
  2. Topics $\beta_{1:K}$, where β_k is a distribution over the vocabulary
  3. Topic proportions $\theta_{1:D}$, where $\theta_{d,k}$ is proportion for topic k in document d
  4. Topic assignments $z_{1:D}$, where $z_{d,n}$ is topic assignment of n^{th} word in document d.
- $\beta_{1:K}$ and $\theta_{1:D}$ are assumed to have Dirichlet distributions, a multivariate version of the Beta distribution

Latent Dirichlet Allocation

Mathematical formulation

- The relationship between these variables is expressed as a joint-distribution:

$$
\begin{aligned}
& p\left(\beta_{1:K}, \theta_{1:D}, z_{1:D}, w_{1:D}\right) \\
& = \prod_{i=1}^{K} p\left(\beta_i\right) \prod_{d=1}^{D} p\left(\theta_d\right) \\
& \quad \left(\prod_{n=1}^{N} p\left(z_{d,n} \mid \theta_d\right) p\left(w_{d,n} \mid \beta_{1:K}, z_{d,n}\right)\right)
\end{aligned}
$$

Latent Dirichlet Allocation

Algorithm

- LDA uses Bayesian inference to estimate the parameters
- We use this formula to compute the *posterior distribution* of $\beta_{1:K}$ and $\theta_{1:D}$
- This is a computationally-intensive process and requires some probabilistic short-cuts
  - *Variational* methods find an approximation of the posterior distribution that fits well*

* Note: This approach is an approximation of the posterior obtained by a more computationally-intensive sampling procedure like Hamiltonian Monte Carlo

Latent Dirichlet Allocation

Loading the corpus

Loading a corpus of State of the Union speeches from 1900-2020. Each row represents a paragraph from a speech. 5000 paragraphs are sampled at random, otherwise models take too long to run!

```r
library(tidyverse)
library(tidytext)
set.seed(89540233)

data <- as_tibble(read_csv("../data/sotu_texts.csv")) %>%
    sample_frac(0.5) # Downsampling by 50%
data$index <- 1:dim(data)[1]
```

Latent Dirichlet Allocation

Inspecting the texts

```r
head(data$paragraph, n=3)
```

Latent Dirichlet Allocation

Preprocessing

```r
data(stop_words)

words <- data %>% unnest_tokens(word, paragraph, strip_punct = TRUE) %>
  anti_join(stop_words)

doc.freq <- words %>%
  count(year, word) %>%
  mutate(n = pmax(pmin(n, 1), 0)) %>%
  group_by(word) %>%
  summarize(df = sum(n))

words <- words %>%
    count(index, word) %>%
    left_join(doc.freq) %>%
    filter(df >= 10)
```

Latent Dirichlet Allocation

Constructing a DTM

```
DTM <- words %>%
  cast_dtm(index, word, n)
dim(DTM)
```

Latent Dirichlet Allocation

Training an LDA model using `topicmodels`

We can pass the DTM to the `LDA` function to train a topic model with 27 topics.

```r
#install.packages("topicmodels")
library(topicmodels)
topic_model<- LDA(DTM, k=27, control = list(seed = 10980))
```

LDA analysis based on code from https://www.tidytextmining.com/topicmodeling.html and https://cbail.github.io/SICSS_Topic_Modeling.html.

Optional

Uncomment the second line and run it to load a pre-fitted model.

```
#save.image(file = "../data/sotu_lda.RData")
load(file = "data/sotu_lda.RData") # Uncomment and run this line to loa
```

Latent Dirichlet Allocation

Analyzing topics

We can use the `tidy` function to pull the beta matrix from the model. Each row corresponds to the probability of word *w* in topic *k*. In this case we can see the probabilities of the term "economy".

```r
topics <- tidy(topic_model, matrix = "beta")

topics %>% filter(term == "economy") %>%
    dplyr::select(topic, beta)
```

Latent Dirichlet Allocation

Analyzing topics

Latent Dirichlet Allocation

Analyzing topics

Latent Dirichlet Allocation

Analyzing topics

Latent Dirichlet Allocation

Topic distributions over documents

We can extract the document-topic proportions. Each row corresponds to the proportion of topic i in document j. Note the notation change. This matrix is called *gamma* (γ) here but was referenced as *theta* (θ) in the equation above. Unfortunately this kind of thing happens a lot as notation is used inconsistently.

Latent Dirichlet Allocation

Latent Dirichlet Allocation

Latent Dirichlet Allocation

Latent Dirichlet Allocation

Helper function

This function can be used to find the documents and words with highest weights in a particular topic.

Latent Dirichlet Allocation

Interpreting the results

"Producing an interpretable solution is the beginning, not the end, of an analysis. The solution constructs meaningful categories and generates corpus-level measures (e.g., the percentage of documents in which a given topic is highly represented) and document-level measures (e.g., the percentage of words in each document assigned to each topic) based on these categories. It remains for the analyst to use this information to address the analytic questions that motivated the research. The analyst must also validate the solution by demonstrating that the model is sound and that his or her interpretation is plausible" (DiMaggio et al. 2013: 586).

Latent Dirichlet Allocation

Inspecting topics

Latent Dirichlet Allocation

Exercise

- Run code above to evaluate your assigned topics
- Add your findings to the shared Google Sheet (requires Rutgers login): https://tinyurl.com/topiccoding

Latent Dirichlet Allocation

Validation

- "There is no statistical test for the optimal number of topics or for the quality of a solution" (DiMaggio, Nag, and Blei 2013: 582)
- They suggest three forms of validation
  - *Statistical*: Calculating various measures of fit
  - *Semantic*: Hand-coding / close reading of documents
  - *Predictive*: Do events change the prevalence of topics?
- Not all topics will be meaningful, some capture residual "junk" and can be ignored (Karell and Freedman 2019)

Latent Dirichlet Allocation

How does it differ from NLP approaches covered so far?

- Document representation
  - A document is represented as a probability distribution over topics, not just a bag-of-words or an embedding
  - Closest approach is *latent semantic analysis*, where a document is a set of weights over latent dimensions. Indeed, LDA was developed as an extension of LSA.
- Document retrieval
  - We can select documents by topic content rather than keywords
  - Words can be shared by multiple topics, unlike conventional keyword detection
- Document comparisons
  - We can compare documents based on topic content rather than text similarity

Latent Dirichlet Allocation

Extensions of LDA

- LDA is considered the “vanilla” topic model. Subsequent approaches have relaxed some of the assumptions of LDA (Blei, 2012):
  - *Assumption 1*: Documents are treated as bags-of-words
    - Language models can be incorporated to better account for linguistic structure
  - *Assumption 2*: Document order does not matter
    - Dynamic topic models account for how topics can change over time
  - *Assumption 3*: The number of topics, K, is known
    - Bayesian non-parametric topic models discover K during the inference procedure

Structural Topic Modeling

Background

- LDA assumes *topic prevalence* (frequency topic is mentioned) and *topic content* (the words used to discuss a topic) are constant across documents
- STM extends LDA by "allowing for the inclusion of covariates of interest into the prior distributions for document-topic proportions and topic-word distributions" (Roberts et al. 2014).
  - This allows analysis of how topics vary according to other factors, for example the treatment in a survey experiment may alter open responses.

Structural Topic Modeling

Topic prevalence

- Prevalence refers to the frequency distribution of a topic across documents
- As social scientists, we often want to see how a topic varies by some categorical variable of interest
    - Author (person, organization, publisher, political party, etc.)
    - Demographics (age group, gender, race, ethnicity, etc.)
    - Time (day, month, year, decade, etc.)

Structural Topic Modeling

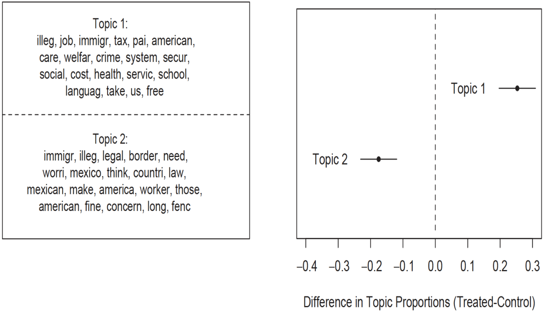
Topic content

- Content refers to the way different topics are discussed
  - Different groups to use different kinds of language to refer to the same topic

Structural Topic Modeling

Analyzing open-ended survey responses using an STM

FIGURE 7 Words and Treatment Effect Associated with Topic 1



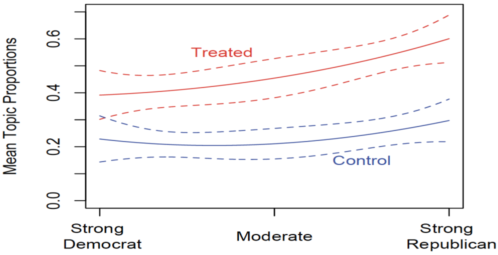
Topic 1:
illeg, job, immigr, tax, pai, american,
care, welfar, crime, system, secur,
social, cost, health, servic, school,
languag, take, us, free

Topic 2:
immigr, illeg, legal, border, need,
worri, mexico, think, countri, law,
mexican, make, america, worker, those,
american, fine, concern, long, fenc

Structural Topic Modeling

Analyzing open-ended survey responses using an STM

FIGURE 8 Party Identification, Treatment, and the Predicted Proportion in Topic 1



Structural Topic Modeling

Selecting metadata

Using year and party from the SOTU corpus as metadata.

```r
meta <- data %>% select(year, party)
head(meta)
```

Structural Topic Modeling

Preprocessing

The `stm` library has its own set of functions for processing data. `textProcessor` takes a corpus, plus metadata, and conducts pre-processing tasks. `prepDocuments` then converts the documents into the appropriate format.

```r
library(stm)
# install.packages("stm")
processed.docs <- textProcessor(data$paragraph,
                                metadata = meta)

output <- prepDocuments(processed.docs$documents,
                        processed.docs$vocab,
                        processed.docs$meta,
                        lower.thresh = 10)
```

Structural Topic Modeling

Finding K

The STM package can calculate some heuristics for finding the optimal value of K. This takes a while as it must run each of the models specified in the vector passed to the K parameter. In this case, the code is running in parallel, each model using a core of the laptop to estimate.

```
library(parallel)
search.results <- searchK(output$documents, output$vocab,
                   K = c(20,40,60,80,100,120),
                   data = output$meta,
                   proportion=0.2, # proportion of docs held-out
                   cores=detectCores() # use maximum number of availabl
)
```

See https://juliasilge.com/blog/evaluating-stm/ for an alternative approach that enables some more post-estimation evaluation.

Structural Topic Modeling

Selecting K

```
plot(search.results)
```

See Mimno, David, Hanna M Wallach, Edmund Talley, Miriam Leenders, and Andrew McCallum. 2011. "Optimizing Semantic Coherence in Topic Models." In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, 262–72. ACL for discussion of the semantic coherence measure.

Structural Topic Modeling

Fitting a model

Fitting a model with `k=60`. The `party` variable is used as a covariate for both prevalence and content. Year is used as a covariate for prevalence, where `s()` is a non-linear spline function.

```
K=60
fit <- stm(documents = output$documents, vocab = output$vocab, K=K,
           data = output$meta,
           prevalence = ~ party + s(year), # s takes a non-linear funct
           content = ~ party, # content can only contain one variable
           verbose = TRUE
           )
```

Structural Topic Modeling

Storing/loading data

I stored the image of this workspace and uploaded it to Github. You can load the trained model and all other files in this script by running this line.

```r
#save.image(file = "../data/sotu_stm.RData")
library(stm)
load(file = "../data/sotu_stm.RData")
```

Structural Topic Modeling

Inspecting topic proportions

We can directly plot the proportions to show how frequent different topics are. Here are the first 20.

```
plot(fit, type = "summary", topics = 1:20)
```

Structural Topic Modeling

Inspecting topic proportions

```
plot(fit, type = "summary", topics = 21:40)
```

Structural Topic Modeling

Inspecting topic proportions

```r
plot(fit, type = "summary", topics =41:60)
```

Structural Topic Modeling

Inspecting topic terms

```
labelTopics(fit, topics=55, n=10)
```

Structural Topic Modeling

Inspecting documents

We can use `findThoughts` to identify documents with a high weight in a given topic. Note that the original texts column does not work, I have to use the index for the metadata file to identify relevant columns.

```
t=55
thoughts <- findThoughts(fit, texts = as.character(data[as.numeric(rown
for (i in unlist(thoughts$docs)) {print(i)}
```

Structural Topic Modeling

Inspecting documents

```r
t=21
thoughts <- findThoughts(fit, texts = as.character(data[as.numeric(rown
for (i in unlist(thoughts$docs)) {print(i)}
```

Structural Topic Modeling

Inspecting documents

```
t=3
thoughts <- findThoughts(fit, texts = as.character(data[as.numeric(rown
for (i in unlist(thoughts$docs)) {print(i)}
```

Structural Topic Modeling

Estimating relationship between topic prevalence and metadata

This function fits a series of models (OLS regressions) to estimate the relationship between topic prevalence and the specified covariates.

```
prep <- estimateEffect(~ party + s(year), fit, meta = output$meta)
```

Structural Topic Modeling

Topic prevalence by party

Structural Topic Modeling

Prevalence over time

We can use the `year` variable to track how prevalence changes over time.

Structural Topic Modeling

Content by party

Structural Topic Modeling

Content by party

Structural Topic Modeling

- Resources
  - The STM website contains information on various tools and research papers that use the approach
    - There are several packages including `stmBrowser`, `stmCorrViz` and `stminsights` that enable more interactive visualization.
  - The vignette provides a closer description of the methodology and a hands-on guide to using the `stm` package.

Alternative approaches

- Several alternatives to STM that may be worth exploring
  - BERT topic models, use embeddings from large language models to produce similar kinds of analyses

    (Egger and Yu 2022)
  - Key word topic models allow topics to be seeded using keywords

    (Eshima, Imai, and Sasaki 2023)
  - Biterm topic models are more suitable for very short texts like social media posts

    (see Greve et al. 2022)

Summary

- Topic modeling is an inductive approach for the summary of large text corpora
    - Analysis of topic models involves the interpretation of topics
    - A key challenge is selecting an appropriate number of topics
- LDA algorithm summarize as corpus into K topics
    - Each document is composed of a mixture of topics
    - Each topic is a mixture of words
- STM improves on LDA by allowing topic prevalence and content to vary by covariates
    - This is particularly useful for social scientific applications