SOSC 4300/5500: Text Analysis; Unsupervised Methods

Han Zhang

Outline

A complete workflow (Grimmer and Stewart, 2013)

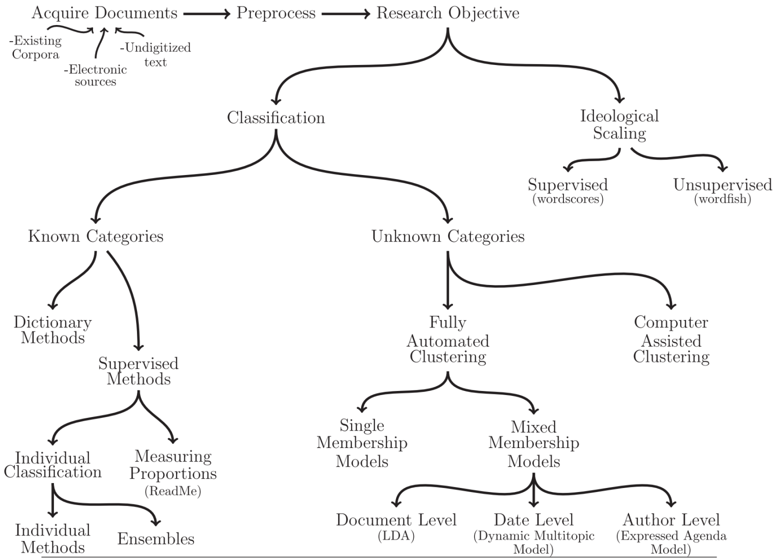


Fig. 1 An overview of text as data methods.

Unsupervised vs supervised

- Supervised: you have a strong *a priori* set of known categories
  - e.g., sentiments, hate speech detection, fake news, election prediction
  - Requires training data to start with for supervised machine learning
- Unsupervised: you do not have a strong *a priori* set of known categories
  - And want the machine to automatically find the categories for you
    - there are risks that the categories found by machines are really not what you want them to be
  - Also called clustering or unsupervised clustering

Setup

- Give you 1 million random tweets, what topics are in these tweets?
- The most basic solution (we will see more complex solution later)
- Transform the texts into numbers
    - document-term matrix
    - embeddings
- Then use some unsupervised methods to group documents into K categories
    - Each document belongs to one category
    - You need to read the documents yourself to give a label to the topic

Unsupervised methods: K-means

- N observations into K clusters
- Each observation belongs to the cluster with the nearest mean (cluster centers, or centroid)
  - Usually the distance metric is Euclidean distances
  - For two observation $x = (x_1, x_2, ..., x_m)$ and $y = (y_1, y_2, ..., y_m)$
  - The euclidean distance is $\sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$
- The k-means algorithm
  - Step 0: select K initial "means" randomly
  - Step 1: associating every observation with the nearest mean
  - Step 2: the centroid of each of the K clusters becomes the new mean.
  - Repeat step 2 and step 3 until convergence

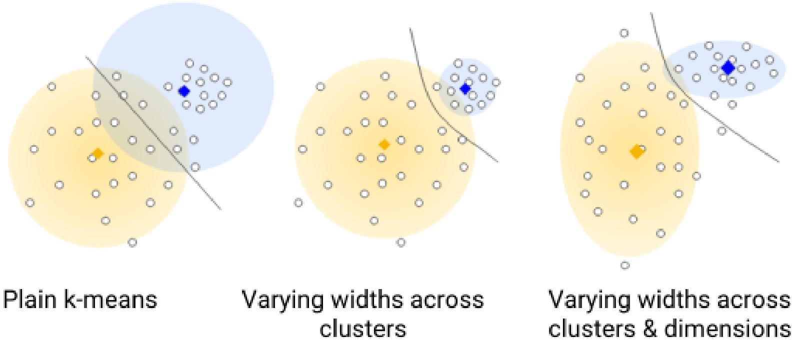
K-means: visualization

- https://www.naftaliharris.com/blog/visualizing-k-means-clustering/
- https://commons.wikimedia.org/wiki/File:K-means_convergence.gif

K-means

- K-means is one of the simplest unsupervised methods -> a good starting point
- But there are shortcomings:
  - Need to select K beforehand
    - This is a general problem for many other clustering methods
  - Sensitive to outliers
  - Sensitive to imbalanced data
  - Does not work very well for high-dimensional data (curse of dimensionality again)

K-means and imbalanced data



- https://developers.google.com/machine-learning/clustering/images/KmeansGeneralization.svg
- Most off-the-shelf packages do not allow you to vary the width of each cluster

K-means and curse of dimensionality

- K-means needs to calculate distance between data and K centroids to find the nearest cluster
- But when the dimension of data is high, the variance between distances decreases
- Then k-means becomes less effective at distinguishing between examples
- `https://developers.google.com/machine-learning/clustering/images/CurseofDimensionality.svg`
- You can perform dimensionality reduction first and then run k-means

Choosing K

- How do you choose K is one of the most challenging part of unsupervised methods
- In supervised methods, we choose parameter values based on whether they improve the prediction performance
- In unsupervised methods, there is no such luck
    - How do I know whether I should choose 5 or 10 clusters?

Choosing K

- Solution 1: Data-driven method
- E.g.., Elbow method
- Inertia: Sum of squared distances of samples to their closest cluster center
  - Problem: bias-variance again; if you let $K = N$, then this distance is 0
- Plot K against inertia
- And you should select a K beyond which the decrease in inertia is not significant
- This is a very heuristic definition and not a hard-rule

Choosing K

https://media.geeksforgeeks.org/wp-content/uploads/20190606105746/inertia.png

Choosing K

- Theory-driven method
  - Just look at $K = 5,6,7,8\ldots$ and which makes most sense to you
- If what you found from Elbow method also makes sense to you, that's the best case

Hierarchical agglomerative clustering

- Bottom-up approach:
  - Each observation starts in its own cluster
  - And pairs of clusters are merged as one moves up the hierarchy.
- Animation: `https://i.gifer.com/80Gy.mp4`
- Pros:
  - No need to specify K
  - Works with imbalanced data well
  - Easy visualization
- Cons:
  - Slow!

Unsupervised methods

- There are many other clustering methods
- Their usage case can be viewed here
- `https://scikit-learn.org/stable/modules/clustering.html`

A complete workflow (Grimmer and Stewart, 2013)

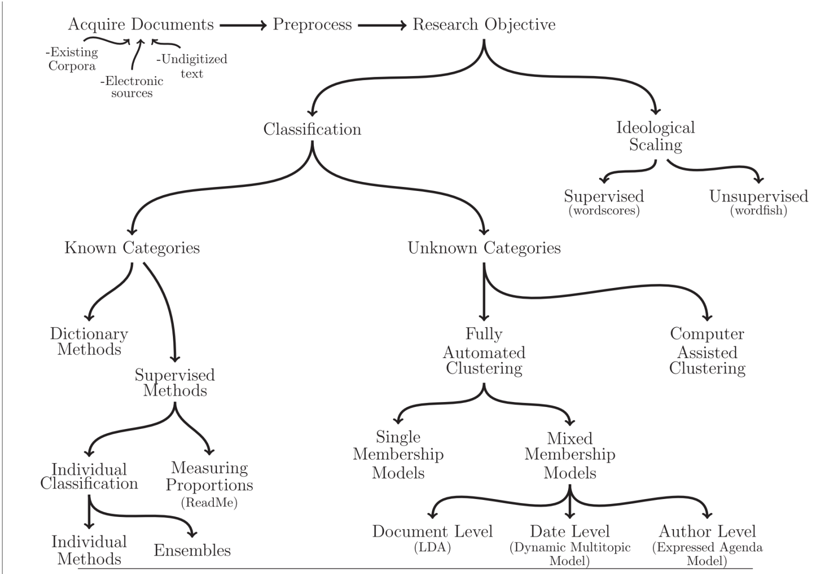


Fig. 1 An overview of text as data methods.

Topic models

- A special type of unsupervised methods designed discovering the main topics for text documents
- Take document-term matrix as the input
- Each document can belong to multiple topics: mixed membership model
- The most basic and the common topic model:
- Latent Dirichlet Allocation (LDA)
- David M. Blei, Andrew Y. Ng, and Michael I. Jordan, *Latent Dirichlet Allocation*, J. Mach. Learn. Res. **3** (2003), 993–1022

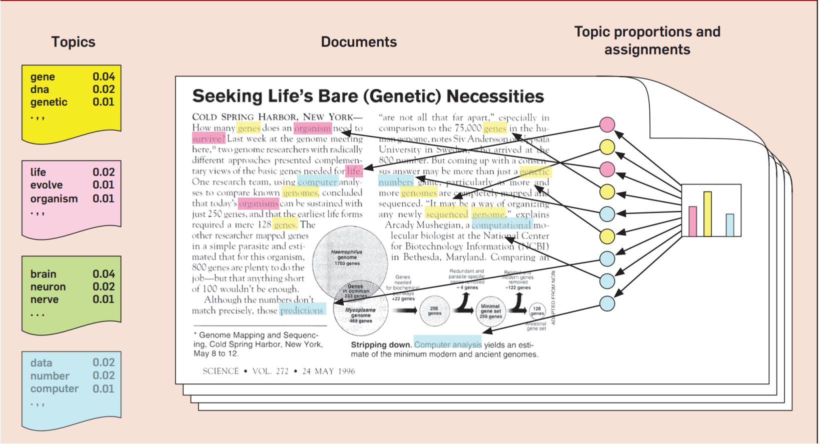
Why not simpler unsupervised methods?

- K-means does not work very well with document-term matrix, which is high-dimensional
- K-means algorithm is assumes single membership:
  - Each document belongs to a single category
  - Not realistic for text documents that often discuss several topics

LDA

David M. Blei, *Probabilistic Topic Models*, Commun. ACM **55** (2012), no. 4, 77–84

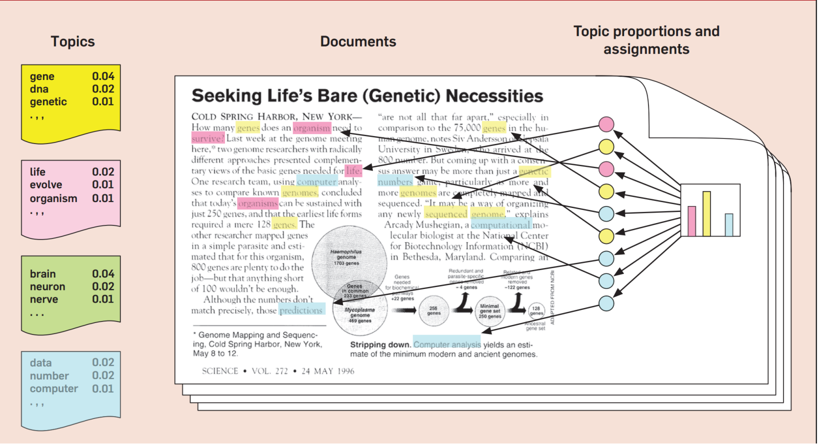
Figure 1. The intuitions behind latent Dirichlet allocation. We assume that some number of "topics," which are distributions over words, exist for the whole collection (far left). Each document is assumed to be generated as follows. First choose a distribution over the topics (the histogram at right); then, for each word, choose a topic assignment (the colored coins) and choose the word from the corresponding topic. The topics and topic assignments in this figure are illustrative—they are not fit from real data. See Figure 2 for topics fit from data.



Documents

- Each document is conceptualized as a probability distribution over topics

Figure 1. The intuitions behind latent Dirichlet allocation. We assume that some number of "topics," which are distributions over words, exist for the whole collection (far left). Each document is assumed to be generated as follows. First choose a distribution over the topics (the histogram at right); then, for each word, choose a topic assignment (the colored coins) and choose the word from the corresponding topic. The topics and topic assignments in this figure are illustrative—they are not fit from real data. See Figure 2 for topics fit from data.



Topics

- Each topic is defined as a probability distribution over words/n-grams
- Mark Steyvers and Tom Griffiths, *Probabilistic topic models*, Handbook of latent semantic analysis **427** (2007), no. 7, 424–440

Topic 247

word	prob.
DRUGS	.069
DRUG	.060
MEDICINE	.027
EFFECTS	.026
BODY	.023
MEDICINES	.019
PAIN	.016
PERSON	.016
MARIJUANA	.014
LABEL	.012
ALCOHOL	.012
DANGEROUS	.011
ABUSE	.009
EFFECT	.009
KNOWN	.008
PILLS	.008

Topic 5

word	prob.
RED	.202
BLUE	.099
GREEN	.096
YELLOW	.073
WHITE	.048
COLOR	.048
BRIGHT	.030
COLORS	.029
ORANGE	.027
BROWN	.027
PINK	.017
LOOK	.017
BLACK	.016
PURPLE	.015
CROSS	.011
COLORED	.009

Topic 43

word	prob.
MIND	.081
THOUGHT	.066
REMEMBER	.064
MEMORY	.037
THINKING	.030
PROFESSOR	.028
FELT	.025
REMEMBERED	.022
THOUGHTS	.020
FORGOTTEN	.020
MOMENT	.020
THINK	.019
THING	.016
WONDER	.014
FORGET	.012
RECALL	.012

Topic 56

word	prob.
DOCTOR	.074
DR.	.063
PATIENT	.061
HOSPITAL	.049
CARE	.046
MEDICAL	.042
NURSE	.031
PATIENTS	.029
DOCTORS	.028
HEALTH	.025
MEDICINE	.017
NURSING	.017
DENTAL	.015
NURSES	.013
PHYSICIAN	.012
HOSPITALS	.011

Figure 1. An illustration of four (out of 300) topics extracted from the TASA corpus.

Word Polysemy

- Each word can belong to multiple topics
  - It's hard to achieve this with dictionary methods
- Mark Steyvers and Tom Griffiths, *Probabilistic topic models*, Handbook of latent semantic analysis **427** (2007), no. 7, 424–440

Topic 77

word	prob.
MUSIC	.090
DANCE	.034
SONG	.033
PLAY	.030
SING	.026
SINGING	.026
BAND	.026
PLAYED	.023
SANG	.022
SONGS	.021
DANCING	.020
PIANO	.017
PLAYING	.016
RHYTHM	.015
ALBERT	.013
MUSICAL	.013

Topic 82

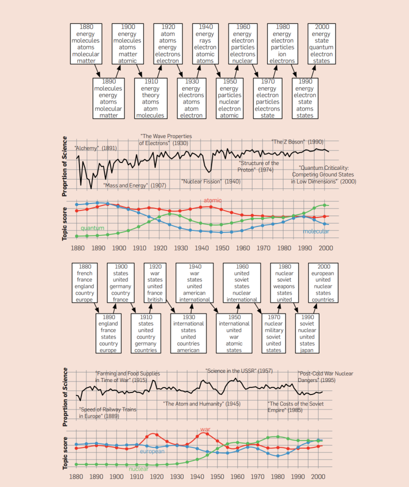
word	prob.
LITERATURE	.031
POEM	.028
POETRY	.027
POET	.020
PLAYS	.019
POEMS	.019
PLAY	.015
LITERARY	.013
WRITERS	.013
DRAMA	.012
WROTE	.012
POETS	.011
WRITER	.011
SHAKESPEARE	.010
WRITTEN	.009
STAGE	.009

Topic 166

word	prob.
PLAY	.136
BALL	.129
GAME	.065
PLAYING	.042
HIT	.032
PLAYED	.031
BASEBALL	.027
GAMES	.025
BAT	.019
RUN	.019
THROW	.016
BALLS	.015
TENNIS	.011
HOME	.010
CATCH	.010
FIELD	.010

Topic changes over time

Figure 5. Two topics from a dynamic topic model. This model was fit to *Science* from 1880 to 2002. We have illustrated the top words at each decade.



LDA algorithm

- Imagine how a computer writes a document with 5,000 words
  1. choose a topic according to the topic distribution (e.g., 0.8 prob of *economy* and 0.2 prob of *politics*)
  2. choose a word according to the topic's word distribution
  3. repeat Step 1 and 2 until you have selected 5,000 words

LDA algorithm

Mark Steyvers and Tom Griffiths, *Probabilistic topic models*, Handbook of latent semantic analysis **427** (2007), no. 7, 424–440

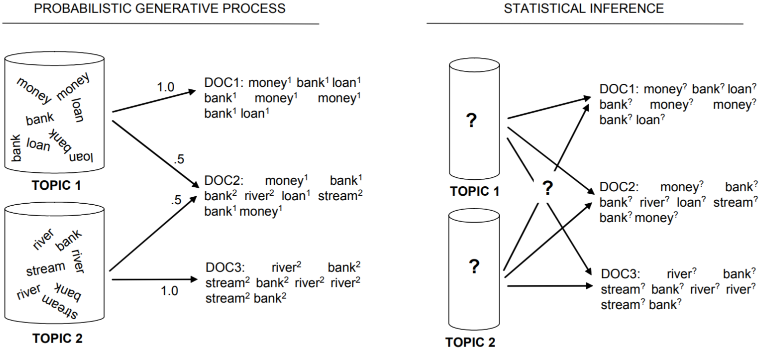


Figure 2. Illustration of the generative process and the problem of statistical inference underlying topic models

LDA algorithm

- *Advanced topic*; it's okay if you do not fully get contents on this slide
- Every ML we taught you is discriminate statistical model
  - regression, LASSO, tree, forests, SVM
- LDA is a generative statistical model
- Differences in math:
  - Discriminative model directly maps features to outcome: $P(Y|X)$
    - E.g., linear regression
    - $p(Y \mid X) = N\left(\beta X, \sigma^2\right)$, where N is a standard normal distribution
- For unsupervised clustering, we do know know Y yet!
- Generative model: model $P(X|Y)$ instead, using Bayes' rule
  - Assumes that we know $P(Y)$
  - And it is easy to calculate $P(X|Y)$

Math of LDA

- *Advanced topic*; it's okay if you do not fully get contents on this slide
- Here, topics are latent outcome Z; and W is the document-term matrix
- A discriminative model will directly model $P(Z|W)$:
    - given document-term matrix (observed)
    - infer topics (unobserved; what we want to obtain)

Math of LDA

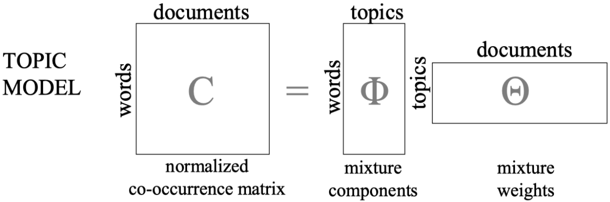
- What LDA does: revert the thinking
-

$$p(w_{id}) = \sum_{j=1}^{K} P(w_i|Z_i = j)P(z_i = j)$$

- $p(w_{id})$ is the probability of observing word i in document d
- $P(w_{id}|Z_d = j) = \phi^j$: word probability in topic j
  - Given topic, what word we should choose?
- $P(Z_d = j) = \theta^d$: topic j's probability of document d
  - Topic's probability itself

Matrix version of LDA

- *Advanced topic*; it's okay if you do not fully get contents on this slide
- A different way to look at LDA is that it decompose the document-term matrix into the product of the following two:
  - term-topic matrix Φ
    - each element is the ϕ in the previous slide)
  - topic-document matrix Θ
    - each element is the θ in the previous slide)
- Looks familiar? LDA is insired by non-negative matrix factorization



Choosing the number of topics

- Topic models: easy to run because no labels needed, but requires significant care in validation
- Choosing K is "one of the most difficult questions in unsupervised learning" (Grimmer and Stewart, 2013, p.19)

Choosing the number of topics

- Two general approaches of choosing parameter values in LDA
- Data-driven method:
  - Still have a hold-out test dataset (like supervised methods)
  - But predict the observed document-term matrix (without labels) for the held-out test data
- perplexity: the original metric used in Blei et al., 2003
  - Use training data to calculate $P(w)$; then calculate likelihood of observing the entire test data over every possible words
  - More words -> lower probability; need weighting
  - It's the inverse probability of the validation set, normalized by the number of words
  - $$PP(W) = p(w_1 w_2 \cdots w_V)^{-1/n} = \sqrt[n]{\frac{1}{p(w_1 \cdots w_V)}}$$
  - The lower the preplexity, the better the model
- Cross-validation again:
  - choose K that minimizes the perplexity on validation set

Choosing the number of topics

- Jonathan Chang, Sean Gerrish, Chong Wang, Jordan Boyd-Graber, and David M. Blei, *Reading Tea Leaves: How Humans Interpret Topic Models*, NIPS 2009
  - Human often disagree with the model chosen by reducing perplexity
- Substantive fit: human in the loop; requires domain-knowledge

Human-Validations

- Grimmer and Stewart, 2013
- Semantic validity:
  - Do the topics identify coherent groups of documents?
- Convergent/discriminant validity
  - Do the topics match existing measures where they should match?
  - Do they depart from existing measures where they should depart?
- Predictive validity
  - Does variation in topic usage correspond with expected events?

Semantic validity

- Chang et al., 2009
- Word intrusion:
  1. select 5 words with the highest probabilities in a topic
  2. select another word that has a low probability in the topic, but high prob in other topics. This is an *intruder* word.
  3. present the 6 words to a human coder, and see if he/she can easily picks up the intruder word.
- e.g., {dog, cat, horse, apple, pig, cow}
  - Easily see that *apple* is an intruder
  - because {dog, cat,horse, pig, cow} make sense together as an animal topic
- You can compare two models on their word intrusion scores

Convergent validity

- Give each topic a label/description
  - which itself is not an easy task
- Ask human coders to read a sample of document and assign them a label
  - but do not show them words in a topic, of course
- And see if the human coding agrees with topic modeling results

Table 4 An example of topic labeling

Description	*Discriminating words*
Iraq War	Iraq, iraqi, troop, war, sectarian
Honorary	Honor, prayer, remember, fund, tribute
Fire Department Grants	Firefight, homeland, afgp, award, equipment

Predictive validity

- Does variation in topic usage correspond with expected events?

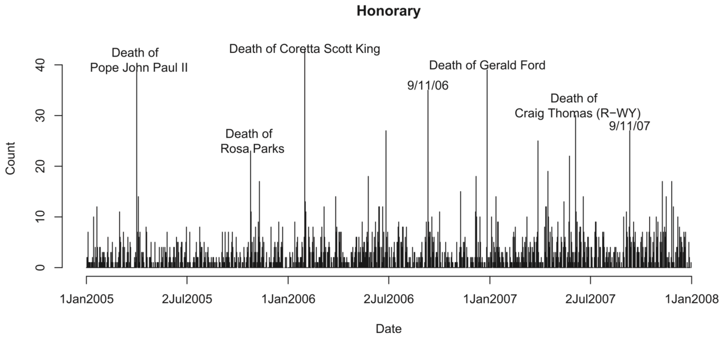


Fig. 4 Predictive validity of topics.

Set up

- Pablo Barberá, Andreu Casas, Jonathan Nagler, Patrick J. Egan, Richard Bonneau, John T. Jost, and Joshua A. Tucker, *Who Leads? Who Follows? Measuring Issue Attention and Agenda Setting by Legislators and the Mass Public Using Social Media Data*, American Political Science Review **113** (2019), no. 4, 883–901
- Questions:
  - Do Congressmen follow issues raised by the public
  - Or, public follows issues raised by Congressmen?
- To answer these questions, we need to measure attention to issues of political discussions

Data

TABLE 1. Description of the Tweets in the Dataset

Group	*N*	Avg	Min	Max	Tweets
House Republicans	238	1,215	70	8,857	267,311
House Democrats	207	1,177	113	5,993	222,491
Senate Republicans	46	1,532	73	6,627	67,412
Senate Democrats	56	1,616	150	10,736	87,307
Random sample	25k	465	1	8,926	11,316,396
Informed public	10k	948	100	5,861	9,487,382
Republican supporters	10k	1,091	100	8,804	10,911,813
Democratic supporters	10k	1,306	100	5,122	13,058,947
Media outlets	36	7,803	8	15,858	273,121

Note: Period of analysis: January 1, 2013, to December 31, 2014. *N* corresponds to the number of Twitter accounts in each sample. *Avg*, *Min*, and *Max* correspond to the average, minimum, and maximum number of tweets, respectively, sent by individual users in each group during the whole period of analysis. *Tweets* corresponds to the total number of tweets sent by all users in each group during the period of analysis.

Topic modeling

- Use LDA to identify issues: each issue is a topic
- Choose 100 topics; based on minimizing perplexity (data-driven approach)
- Why they do not choose supervised methods? What are their arguments?
    1. too many categories; requires too many labeled documents
        - Say 500 documents per each category; that is 50,000; not a small number
    2. they do passed the convergent validity check

Topics

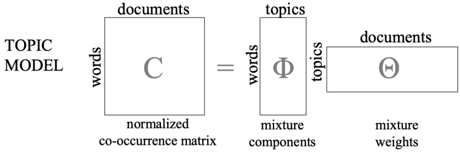
TABLE 2. List of Political Issues

Topic Number	Label	Topic number	Label
3	Investigation of Benghazi attack	50	Climate change
7	100 days of #BringBackOurGirls campaign	51	Lame duck congress
9	Gender wage gap	53	Minimum wage
12	Republican issues *Spring 2013*	58	Affordable Care Act
14	Marriage equality	62	Border crisis in Texas
15	Gun violence	63	Obamacare (employer mandate)
16	Abortion (pro-life)	64	FAA furloughs cause flight delays
18	Veteran affairs delays scandal	66	Malaysia Airlines crash in Ukraine
20	NSA surveillance scandal	67	Comprehensive immigration reform
23	#BringBackOurGirls campaign	70	#MiddleClassFirst campaign
28	Employment Non-Discrimination Act	75	Military Justice Improvement Act
32	Islamic state	81	Poverty (SNAP program)
33	Use of military force in Syria	83	Twenty-first century cures initiative
36	Ebola	85	Unemployment insurance
37	Social security	88	IRS scandal
39	Keystone XL pipeline	89	Obamacare (website and implementation)
41	Immigration (border security)	93	Jobs bills omnibus
43	Executive action on immigration	96	Violence Against Women Act
46	Unemployment numbers reports	97	Protests in Ukraine and Venezuela
47	Paul Ryan budget proposal	99	CIA detentions and interrogations report
48	Black history month	100	#ObamacareInThreeWords campaign
(101)	Student debt	(102)	Hobby lobby supreme court decision
(103)	Budget discussion	(104)	2013 government shutdown

Note: The topic number in parentheses indicate issues that have been created *ad hoc* by merging very similar topics from the topic model.

Measure attention to issues

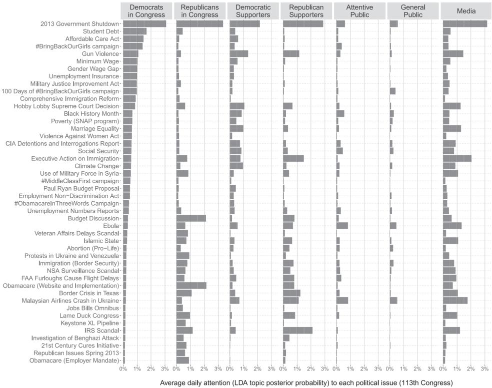
- After getting a list of issues
- They measure attention to issues as the daily posterior LDA probabilities for each group
  - They basically mean $P(Z_d = j) = \theta^d$: topic j's probability of document d
- Or the column means of the Θ matrix (topic-document matrix)



Measure attention to issues

- Result: Y_{ijt}; proportion of topic i for group j at time t

FIGURE 1. Average Issue Attention by Groups of Politicians, the Public, and the Media



Statistical analysis

- Topic modeling help the authors to obtain the key independent and dependent variables Y_{ijt}
- To answer their questions:
  - Do Congressmen follow issues raised by the public
  - Or, public follows issues raised by Congressmen?
- A series of vector auto regression regressions (a standard model dealing with time-series data)
  - Basically, can Y_{ijt} can predict $Y_{i,j',t+1}$?
  - Or in plain language, can public's issue attention predict Congressmen's future issue attention
  - Or vice versa

Revisiting unsupervised vs supervised methods

- Unsupervised methods (especially topic modeling) are widely used
  - Because they allows you to explore the corpus, without the no need gather thousands of training documents (practical consideration)
- But categories found by unsupervised methods are not necessarily what you expect
  - E.g., if you use topic model, it's very likely that the topics returned are mixed with both positive and negative sentiments
- You probably can try unsupervised methods first
  - if the categories unsupervised methods found do not suit your need, shift to supervised methods

Using embedding in topic modeling

- K-means does not work very well with document-term matrix, which is high-dimensional
- LDA takes document-matrix as input, and solves the high-dimensional problem
- How about we use K-means on embeddings, not on document-term matrix?
- It's working pretty well

Steps

Maarten Grootendorst, *BERTopic: Neural topic modeling with a class-based TF-IDF procedure*, 2022

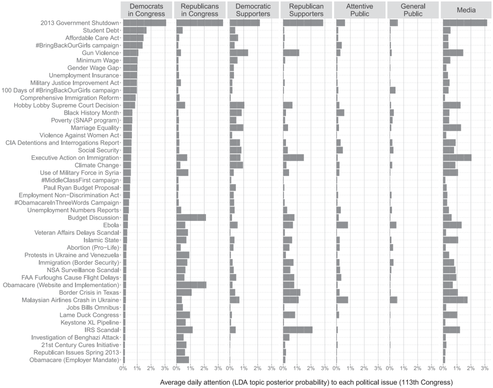
- `https://maartengr.github.io/BERTopic/index.html`
- embed each document as a vector (default dimension $= 768$)
- use UMAP (another dimensional reduction method) to further reduce data dimensions to around 5 - 10
  - PCA should also be fine
- Apply K-means or other clustering algorithms
- Select top words (i.e., representative words of a topic) using tf-idf scores

What to do with covariates in topic models?

- LDA only finds topics
- Suppose you have covariates, and want to see how topics vary by covariates
  - That's a central question in Barbera et. al, (2019) we read in last week
  - E.g., whether a topic's proportions change over time?
  - whether a topic's proportion changes by type of authors?

Example: Barbera et. al, (2019)

FIGURE 1. Average Issue Attention by Groups of Politicians, the Public, and the Media



Problems

- Simple workarounds: split the documents by year/author type, and fit LDA separately for subsets
- Problems?
  - Topics are not comparable
  - Normal people's political discussions may be very different from that of politicians
  - Certain issues may only be discussed by politicians, but not by politicians, or vice versa
- How did Barbera solved this problem? Anyone remembers?
  - They fit LDA for politicians only, thus effectively fixing the allowed topics.
  - And of course, they sacrifice the possibility that normal people may come up with entirely different topics that's not in politician's discussions

Structure Topic Model (STM)

- Margaret E. Roberts, Brandon M. Stewart, Dustin Tingley, Christopher Lucas, Jetson Leder-Luis, Shana Kushner Gadarian, Bethany Albertson, and David G. Rand, *Structural Topic Models for Open-Ended Survey Responses*, American Journal of Political Science **58** (2014), no. 4, 1064–1082
- STM is particularly popular among social scientists
- One reason is that it's fairly easy to use; with a R package provided by the authors
  - Most of Blei's models are implemented in C/C++; not even in Python. Not friendly to social scientists
- The other reason is that they tried to combine LDA with regressions, which social scientists are familiar with
  - CS people do not care about regressions, of course

STM: topical prevalence

- What LDA does:
  - $$p(w_{id}) = \sum_{j=1}^{K} P(w_i|Z_i = j)P(z_i = j)$$
  - $p(w_{id})$ is the probability of observing word i in document d
  - $P(w_{id}|Z_d = j) = \phi^j$: probability of observing word i in document d, if we know d's topic is j
  - $P(Z_d = j) = \theta^d$: topic j's probability of document d
- How STM extends?
  - $P(Z_d = j) = \theta^d = \mathit{LogisticNormal}(\beta X, \epsilon)$
    - That is, topic's probability of document d can vary by covariates of d, X_d
    - In STM's notation, covariates can influence topical prevalence
    - Recall that linear regression looks like $P(Y|X) = \mathit{Normal}(\beta X_d, \epsilon)$
    - LogisticNormal extends Normal distribution, by taking the logic transformation and forcing values to be between (0,1), which is required (since we are modeling probabilities!)

STM: topical contents

- What LDA does:
  - $$p(w_{id}) = \sum_{j=1}^{K} P(w_i|Z_i = j)P(z_i = j)$$
  - $p(w_{id})$ is the probability of observing word i in document d
  - $P(w_{id}|Z_d = j) = \phi^j$: probability of observing word i in document d, if we know d's topic is j
  - $P(Z_d = j) = \theta^d$: topic j's probability of document d
- How STM extends?
  - $P(w_{id}|Z_d = j) = \phi^j + \kappa_{X_d}$
    - That is, probability of using word i in document d varies by document's covariates
    - In STM's notation, covariates can influence topical content

Prevalence vs. Content

- If you have some covariates X (e.g., year, politician/normal people)
- You do not need to let them influence prevalence and content simultaneously
- Allow X to determine topical prevalence if:
  - You can about how topics vary by X
- Allow X to determine topical if:
  - You want to see how word usage varies by X, within a topic