Topic Models

Kenneth Benoit & Pablo Barberá

MY 459: Quantitative Text Analysis

March 12th, 2018

Course website: lse-my459.github.io

Topic Models

- Topic models are algorithms for discovering the main "themes" in an unstructured corpus
- Requires no prior information, training set, or special annotation of the texts
  – only a decision on K (number of topics)
- A probabalistic, generative advance on several earlier methods, "Latent Semantic Analysis" (LSA) and "probabalistic latent semantic indexing" (pLSI)

differences from previous models

unigram model each word each word is assumed to be drawn from the same term distribution

mixture of unigram models a topic is drawn for each document and all words in a document are drawn from the term distribution of the topic

mixed-membership models documents are not assumed to belong to single topics, but to simultaneously belong to several topics and the topic distributions vary over documents

Uses and applications

- Topic models are algorithms for discovering the main themes that pervade a large and otherwise unstructured collection of documents
- Can be used to organize the collection according to the discovered themes
- Topic modeling algorithms can be applied to massive collections of documents
- Topic modeling algorithms can be adapted to many kinds of data. among other applications, they have been used to find patterns in genetic data, images, and social networks

Advantages over cruder methods

- parametric, so we get estimates of parameters for topic proportions in each document, and topic weights for each word
- can incorporate additional information hierarchically (e.g. using "structural" topic models)
- but we pay for these benefits in the form of far greater computational complexity

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.

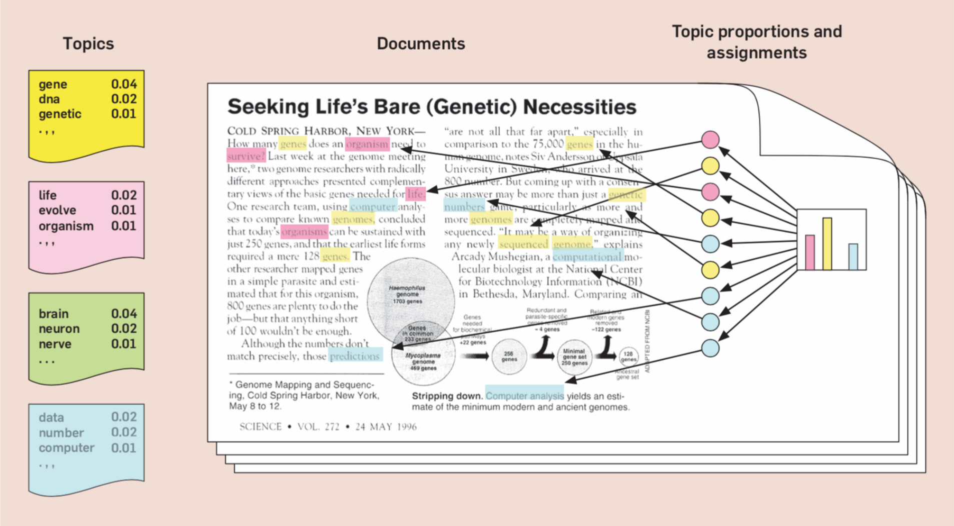
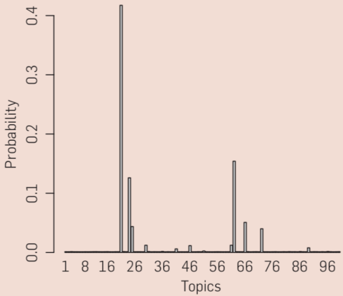


Figure 2. Real inference with LDA. We fit a 100-topic LDA model to 17,000 articles from the journal *Science*. At left are the inferred topic proportions for the example article in Figure 1. At right are the top 15 most frequent words from the most frequent topics found in this article.



"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

Latent Dirichlet Allocation: Overview

- The LDA model is a Bayesian mixture model for discrete data where topics are assumed to be uncorrelated (in “classic” LDA)
- LDA provides a generative model that describes how the documents in a dataset were created
- Each of the K *topics* is a distribution over a fixed vocabulary
- Each document is a collection of words, generated according to a multinomial distribution, one for each of K topics
- Inference consists of estimating a posterior distribution from a joint distribution based on the probability model from a combination of what is observed (words in documents) and what is hidden (topic and word parameters)

Illustration of the LDA generative process

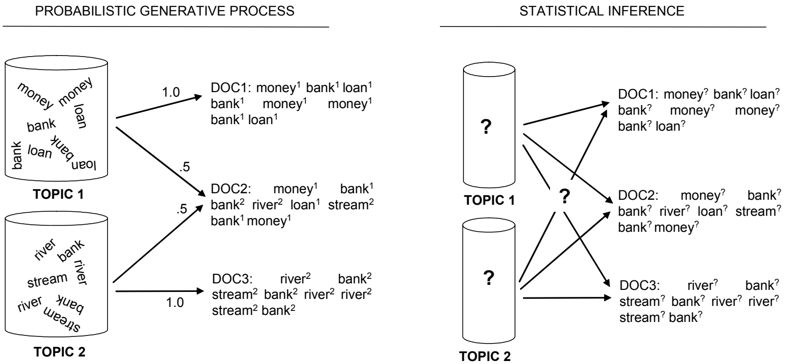


Figure 2. Illustration of the generative process and the problem of statistical inference underlying topic models

(from Steyvers and Griffiths 2007)

Topics example

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

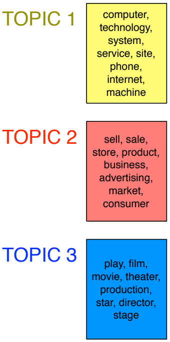
word	prob.
DOCTOR	.074
DR.	.063
PATIENT	.061
HOSPITAL	.049
CARE	.046
MEDICAL	.042
NURSE	.031
PATIENTS	.029
DOCTORS	.028
HEALTH	.025
MEDICINE	.017
NURSING	.017
DENTAL	.015
NURSES	.013
PHYSICIAN	.012
HOSPITALS	.011

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

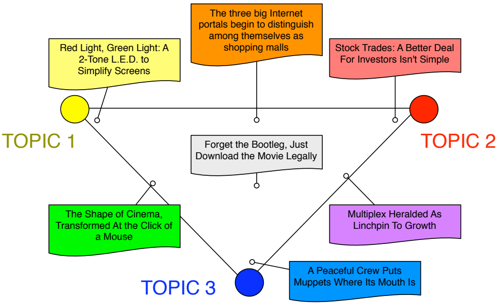
(from Steyvers and Griffiths 2007)

Often K is quite large!

Example



(a) Topics



(b) Document Assignments to Topics

Latent Dirichlet Allocation: Details

- Document = random mixture over latent topics
- Topic = distribution over n-grams

Probabilistic model with 3 steps:

1. Choose $\theta_i \sim \mathrm{Dirichlet}(\alpha)$
2. Choose $\beta_k \sim \mathrm{Dirichlet}(\delta)$
3. For each word in document *i*:
   - Choose a topic $z_m \sim \mathrm{Multinomial}(\theta_i)$
   - Choose a word $w_{im} \sim \mathrm{Multinomial}(\beta_{i,k=z_m})$

where:

α=parameter of Dirichlet prior on distribution of topics over docs.
θ_i=topic distribution for document *i*
δ=parameter of Dirichlet prior on distribution of words over topics
β_k=word distribution for topic *k*

Latent Dirichlet Allocation

Key parameters:

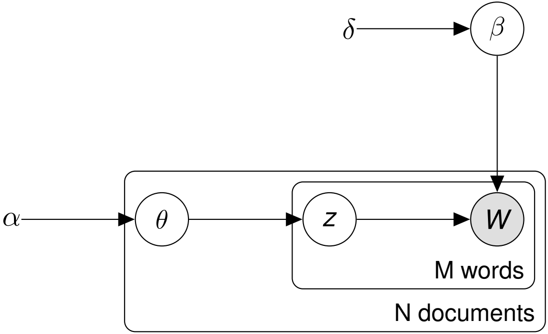
1. θ = matrix of dimensions N documents by K topics where θ_{ik} corresponds to the probability that document i belongs to topic k; i.e. assuming $K = 5$:

	T1	T2	T3	T4	T5
Document 1	0.15	0.15	0.05	0.10	0.55
Document 2	0.80	0.02	0.02	0.10	0.06
...					
Document N	0.01	0.01	0.96	0.01	0.01

2. β = matrix of dimensions K topics by M words where β_{km} corresponds to the probability that word m belongs to topic k; i.e. assuming $M = 6$:

	W1	W2	W3	W4	W5	W6
Topic 1	0.40	0.05	0.05	0.10	0.10	0.30
Topic 2	0.10	0.10	0.10	0.50	0.10	0.10
...						
Topic k	0.05	0.60	0.10	0.05	0.10	0.10

Plate notation



$\beta = M \times K$ matrix where β_{im} indicates prob(topic=k) for word m
$\theta = N \times K$ matrix where θ_{ik} indicates prob(topic=k) for document i

Validation

From Quinn et al, AJPS, 2010:

1. Semantic validity
   - Do the topics identify coherent groups of tweets that are internally homogenous, and are related to each other in a meaningful way?
2. Convergent/discriminant construct validity
   - Do the topics match existing measures where they should match?
   - Do they depart from existing measures where they should depart?
3. Predictive validity
   - Does variation in topic usage correspond with expected events?
4. Hypothesis validity
   - Can topic variation be used effectively to test substantive hypotheses?

Example: open-ended survey responses

Bauer, Barberá *et al*, *Political Behavior*, 2016.

- Data: General Social Survey (2008) in Germany
- Responses to questions: *Would you please tell me what you associate with the term "left"? and would you please tell me what you associate with the term "right"?*
- Open-ended questions minimize priming and potential interviewer effects
- Sparse Additive Generative model instead of LDA (more coherent topics for short text)
- $K = 4$ topics for each question

Example: open-ended survey responses

Table 1: Top scoring words associated with each topic, and English translations)

Left topic 1: **Parties** (proportion = .26, average lr-scale value = 5.38) linke, spd, partei, linken, pds, politik, kommunisten, parteien, grünen, punks *the left, spd, party, the left, pds, politics, communists, parties, greens, punks*
Left topic 2: **Ideologies** (proportion = .26, average lr-scale value = 5.36) kommunismus, links, sozialismus, lafontaine, rechts, aber, gysi, linkspartei, richtung, gleichmacherei *communism, left, socialism, lafontaine, right, but, gysi, left party, direction, levelling*
Left topic 3: **Values** (proportion = .24, average lr-scale value = 4.06) soziale, gerechtigkeit, demokratie, soziales, bürger, gleichheit, gleiche, freiheit, rechte, gleichberechtigung *social, justice, democracy, social, citizen, equality, equal, freedom, rights, equal rights*
Left topic 4: **Policies** (proportion = .24, average lr-scale value =4.89) sozial, menschen, leute, ddr, verbinde, kleinen, einstellung, umverteilung, sozialen, vertreten *social, humans, people, ddr, associate, the little, attitude, redistribution, social, represent*

Right topic 1: **Ideologies** (proportion = .27, average lr-scale value = 5.00) konservativ, nationalsozialismus, rechtsradikal, radikal, ordnung, politik, nazi, recht, menschen, konservative *conservative, national socialism, right-wing radicalism, radical, order, politics, nazi, right, people, conservatives*
Right topic 2: **Parties** (proportion = .25, average lr-scale value = 5.26) npd, rechts, cdu, csu, rechten, parteien, leute, aber, verbinde, rechtsradikalen *npd, right, cdu, csu, the right, parties, people, but, associate, right-wing radicalists*
Right topic 3: **Xenophobia** (proportion = .25, average lr-scale value = 4.55) ausländerfeindlichkeit, gewalt, ausländer, demokratie, nationalismus, rechtsradikalismus, diktatur, national, intoleranz, faschismus *xenophobia, violence, foreigners, democracy, nationalism, right-wing radicalism, dictatorship, national, intolerance, fascism*
Right topic 4: **Right-wing extremists** (proportion = .23, average lr-scale value = 4.90) nazis, neonazis, rechtsradikale, rechte, radikale, radikalismus, partei, ausländerfeindlich, reich, nationale *nazis, neonazis, right-wing radicalists, rightists, radicals, radicalism, party, xenophobia, rich, national*

Note: "proportion" indicates the average estimated probability that any given response is assigned to a topic. "average lr-scale value" is the mean position on the left-right scale (from 0 to 10) of individuals whose highest probability belongs to that particular topic.

Bauer, Barberá *et al*, *Political Behavior*, 2016.

Example: open-ended survey responses

Fig. 6: Left-right scale means for different subsamples of associations with **left** (dashed = sample mean, bars = 95% Cis)

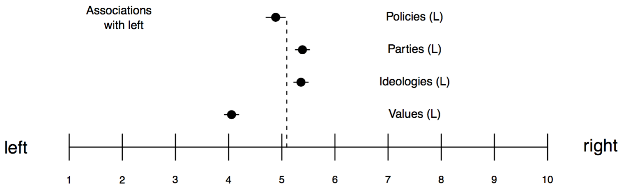
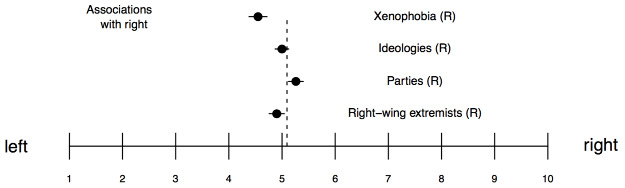


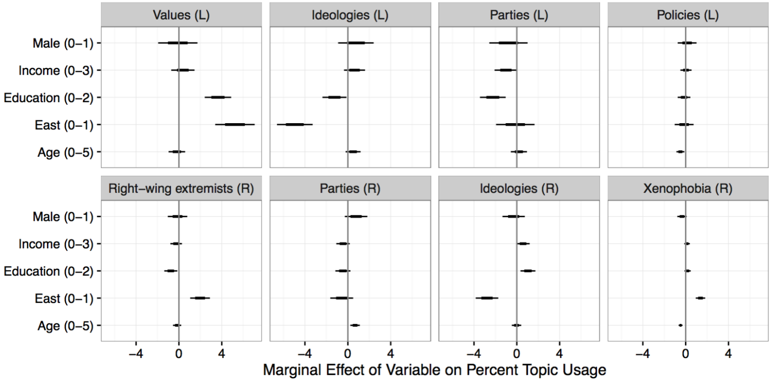
Fig. 7: Left-right scale means for different subsamples of associations with **right** (dashed = sample mean, bars = 95% Cis)



Bauer, Barberá *et al*, *Political Behavior*, 2016.

Example: open-ended survey responses

Fig. 9: Systematic relationship between associations with “left” and “right” and characteristics of respondents



Note: Each line indicates a 95% confidence interval (and 66% confidence interval in darker color) for the coefficient of eight different regressions of topic usage (in a scale from 0 to 100) at the respondent level on seven individual-level characteristics. The line on the bottom right corner (second row, second plot), for example, shows that individual a one-category change in age is associated with around one percentage point increase in the probability that the individual associated “right” with political parties.

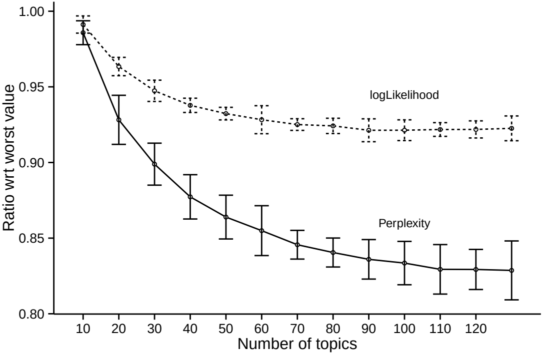
Bauer, Barberá *et al*, *Political Behavior*, 2016.

Example: topics in US legislators' tweets

- Data: 651,116 tweets sent by US legislators from January 2013 to December 2014.
- 2,920 documents = 730 days $\times$ 2 chambers $\times$ 2 parties
- Why aggregating? Applications that aggregate by author or day outperform tweet-level analyses (Hong and Davidson, 2010)
- $K = 100$ topics (more on this later)
- Validation: http://j.mp/lda-congress-demo

Choosing the number of topics

- Choosing K is "one of the most difficult questions in unsupervised learning" (Grimmer and Stewart, 2013, p.19)
- We chose $K = 100$ based on cross-validated model fit.



Choosing the number of topics (contd.)

- **BUT**: “there is often a negative relationship between the best-fitting model and the substantive information provided”.
- GS propose to choose K based on “substantive fit.”

Model evaluation using "perplexity"

- can compute a likelihood for "held-out" data
- perplexity: can be computed as (using VEM):

$$\text{perplexity}(w) = \exp\left\{-\frac{\sum_{d=1}^{M} \log p(w_d)}{\sum_{d=1}^{M} N_d}\right\}$$

- lower perplexity score indicates better performance

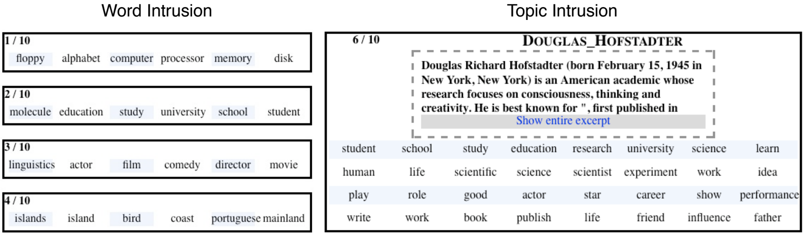
Evaluating model performance: human judgment

(Chang, Jonathan et al. 2009. “Reading Tea Leaves: How Humans Interpret Topic Models.” *Advances in neural information processing systems*.)

Uses human evaluation of:

- whether a topic has (human-identifiable) semantic coherence: word intrusion, asking subjects to identify a spurious word inserted into a topic
- whether the association between a document and a topic makes sense: topic intrusion, asking subjects to identify a topic that was not associated with the document by the model

Example



- conclusions: the quality measures from human benchmarking were negatively correlated with traditional quantitative diagnostic measures!

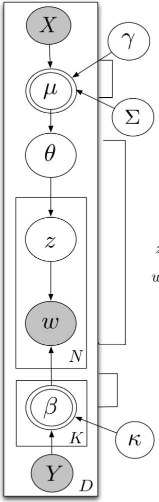
Extensions of LDA

1. Structural topic model (Roberts et al, 2014, AJPS)
2. Dynamic topic model (Blei and Lafferty, 2006, ICML; Quinn et al, 2010, AJPS)
3. Hierarchical topic model (Griffiths and Tenembaun, 2004, NIPS; Grimmer, 2010, PA)

Why?

- Substantive reasons: incorporate specific elements of DGP into estimation
- Statistical reasons: structure can lead to better topics.

Structural topic model



Topic Prevalence:

$$\mu_{d,k} = X_d \gamma_k$$
$$\gamma_k \sim \mathcal{N}(0, \sigma_k^2)$$
$$\sigma_k^2 \sim \text{Gamma}(s^\gamma, r^\gamma)$$

Language Model:

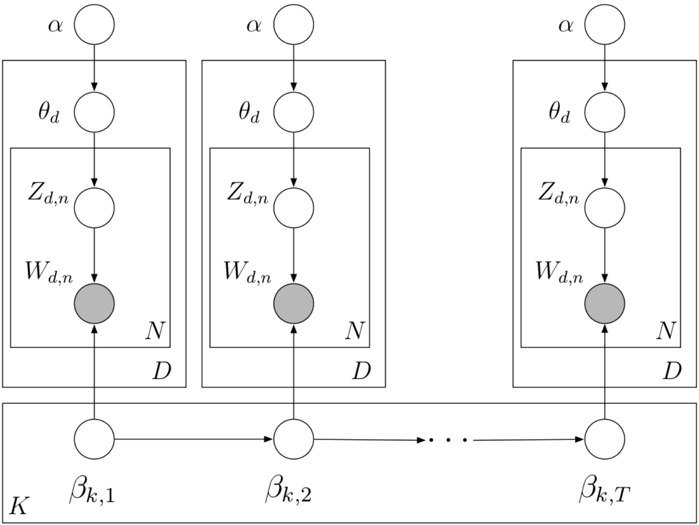
$$\theta_d \sim \text{LogisticNormal}(\mu_d, \Sigma)$$
$$z_{d,n} \sim \text{Mult}(\theta_d)$$
$$w_{d,n} \sim \text{Mult}(\beta_d^{k=z_{d,n}})$$

Topical Content:

$$\beta_{d,v}^k \propto \exp(m_v + \kappa_v^{\cdot,k} + \kappa_v^{y,\cdot} + \kappa_v^{y,k})$$
$$\kappa_v^{y,k} \sim \text{Laplace}(0, \tau_v^{y,k})$$
$$\tau_v^{y,k} \sim \text{Gamma}(s^\kappa, r^\kappa)$$

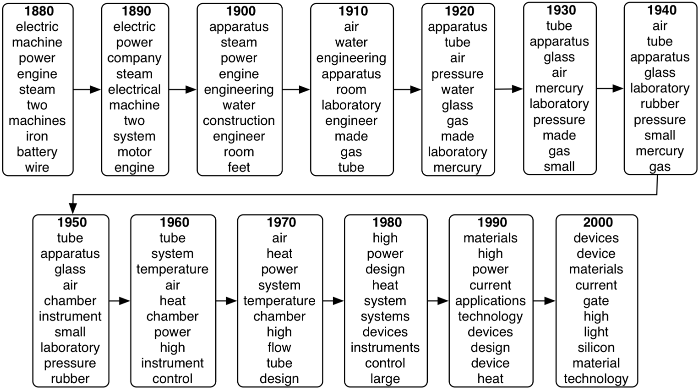
- **Prevalence**: Prior on the mixture over topics is now document-specific, and can be a function of covariates (documents with similar covariates will tend to be about the same topics)
- **Content**: distribution over words is now document-specific and can be a function of covariates (documents with similar covariates will tend to use similar words to refer to the same topic)

Dynamic topic model



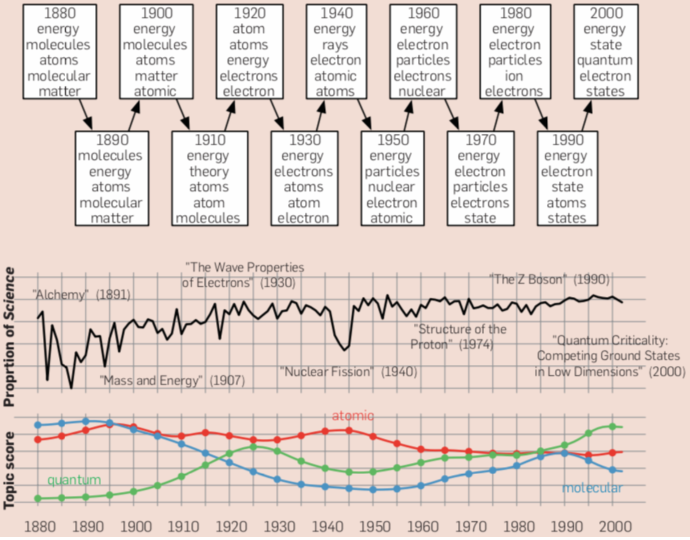
Source: Blei, “Modeling Science”

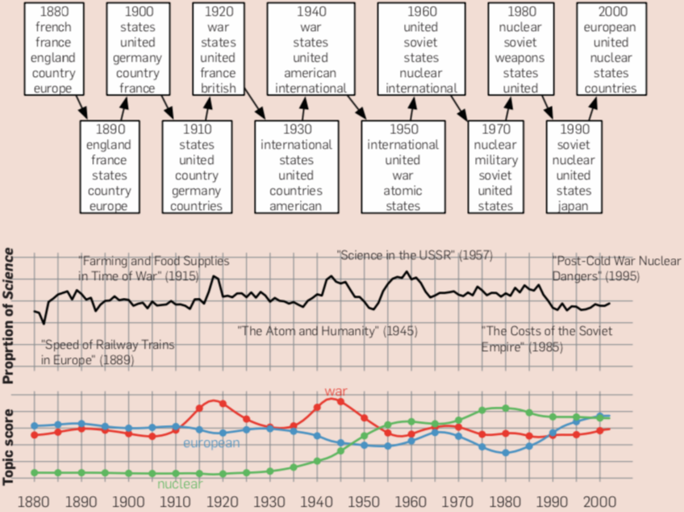
Dynamic topic model



Source: Blei, “Modeling Science”

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.





1880 french france england country europe
1890 england france states country europe
1900 states united germany country france
1910 states united country germany countries
1920 war states united france british
1930 international states united countries american
1940 war states united american international
1950 international united war atomic states
1960 united soviet states nuclear international
1970 nuclear military soviet united states
1980 nuclear soviet weapons states united
1990 soviet nuclear united states japan
2000 european united nuclear states countries
"Farming and Food Supplies in Time of War" (1915)
"Science in the USSR" (1957)
"Post-Cold War Nuclear Dangers" (1995)
"The Atom and Humanity" (1945)
"The Costs of the Soviet Empire" (1985)
"Speed of Railway Trains in Europe" (1889)
Proprtion of Science
Topic score
war
european
nuclear
1880
1890
1900
1910
1920
1930
1940
1950
1960
1970
1980
1990
2000

Drawbacks of LDA

- discards word order
- assumes documents are exchangeable
- the setting of the hyperparameters has led to a great deal of confusion, even as we note above, leading to a misconception about the effective- ness of different forms of posterior inference
- unclear how to choose the number of topics K

Which implementation in R?

- `lda`
- `topicmodels`
- `mallet`
- `stm`

In quanteda, matrices compatibile as inputs for these functions can be created using `convert(x, ...)`