

Day 11: Topic Models

ME314: Introduction to Data Science and Big Data Analytics

Jack Blumenau

14th August 2019

Day 10 Outline

Topic Models

Latent Dirichlet Allocation (LDA)

Beyond Latent Dirichlet Allocation

Correlated and Dynamic Topic Models

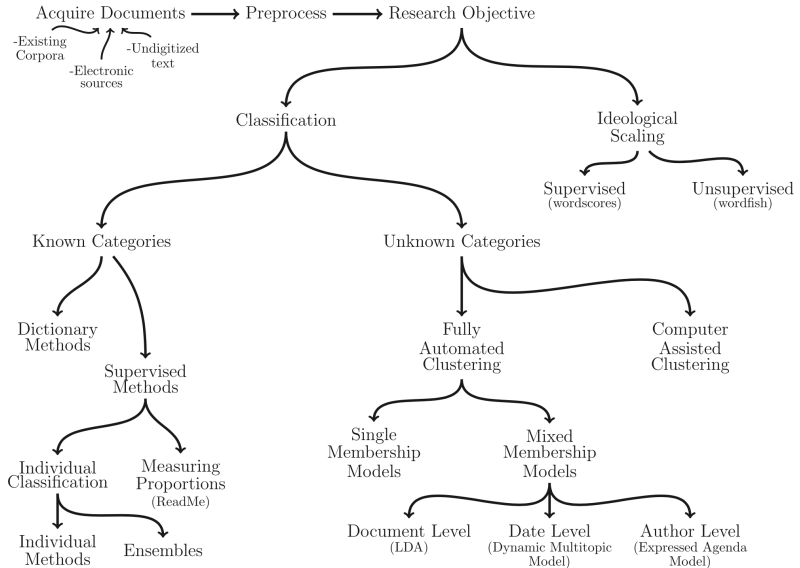
Structural Topic Model

Summary

Recap

- Quantitative text analysis always requires:
 1. Construction of a quantitative matrix from textual features
 2. A quantitative or statistical procedure applied to that matrix
 3. Summary or interpretation of the results of that procedure

Where are we?



Where are we?

- For the past two days:
 1. Dictionary approaches
 2. Supervised approaches
 3. Scaling methods
- Today we move on to unsupervised methods
- Note that we will still need to make many of the same feature selection decisions as we did previously...

Topic Models

Introduction to topic models

Topic models allow us to cluster similar documents in a corpus together.

Wait. Don't we already have tools for that?

Introduction to topic models

Topic models allow us to cluster similar documents in a corpus together.

Wait. Don't we already have tools for that? Yes! Dictionaries and supervised learning.

Introduction to topic models

Topic models allow us to cluster similar documents in a corpus together.

Wait. Don't we already have tools for that? Yes! Dictionaries and supervised learning.

So what do topic models add?

What do topic models add?

		Do you know the categories in which you want to place documents?	
		Yes	No
Do you know the rule for placing documents in categories?	Yes		
	No		

What do topic models add?

		Do you know the categories in which you want to place documents?
		Yes No
Do you know the rule for placing documents in categories?	Yes	Dictionary methods
	No	

What do topic models add?

		Do you know the categories in which you want to place documents?
		Yes No
Do you know the rule for placing documents in categories?	Yes	Dictionary methods
	No	Supervised learning

What do topic models add?

		Do you know the categories in which you want to place documents?	
		Yes	No
Do you know the rule for placing documents in categories?	Yes	Dictionary methods	
	No	Supervised learning	Topic Models

What do topic models add?

		Do you know the categories in which you want to place documents?	
		Yes	No
Do you know the rule for placing documents in categories?	Yes	Dictionary methods	NA
	No	Supervised learning	Topic Models

Introduction to topic models

- Topic models are algorithms for discovering the main themes in an unstructured corpus
- They require no prior information, training set, or labelling of texts before estimation
- They allow us to automatically organise, understand, and summarise large archives of text data.
 1. Uncover hidden themes.
 2. Annotate the documents according to themes.
 3. Organise the collection using annotations.

What is a “topic”?

- **Google:** “a matter dealt with in a text or conversation; a subject.”
- **Topic models:** probability distribution over a fixed word vocabulary
- Consider a vocabulary: gene, dna, genetic, data, number, computer
- When speaking about **genetics**, you will:
 - frequently use the words “gene”, “dna” & “genetic”
 - infrequently use the words “data”, “number” & “computer”
- When speaking about **computation**, you will:
 - frequently use the words “data”, “number” & “computation”
 - infrequently use the words “gene”, “dna” & “genetic”

Topic	gene	dna	genetic	data	number	computer
Genetics	0.4	0.25	0.3	0.02	0.02	0.01
Computation	0.02	0.01	0.02	0.3	0.4	0.25

A motivating example

- Data: UK House of Commons' debates (PMQs)
 - ≈ 30000 parliamentary speeches from 1997 to 2015
 - ≈ 3000 unique words
 - $\approx 2m$ total words
- Note that I have already made a number of sample selection decisions
 - Only PMQs ($\approx 3\%$ of total speeches)
 - Removed frequently occurring & very rare words
 - All words have been “stemmed”
- Results of a 30-topic model

Latent Dirichlet Allocation (LDA)

Latent Dirichlet Allocation (LDA)

Topics

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
...

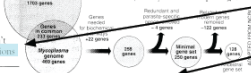
Documents

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, a geneticist at the University of Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a numbers game, particularly if 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 the



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

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

SCIENCE • VOL. 272 • 24 MAY 1996

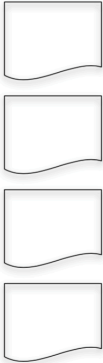
Topic proportions and assignments



- Each topic is a distribution over words
- Each document is a mixture of corpus-wide topics
- Each word is drawn from one of those topics

Latent Dirichlet Allocation (LDA)

Topics



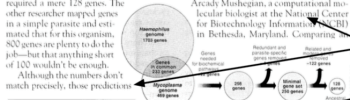
Documents

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 at Uppsala University in Sweden. But coming up with a consensus answer may be more than just a numeric numbers game. Particularly more and more genomes are being rapidly sequenced 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

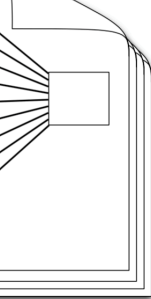


* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

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

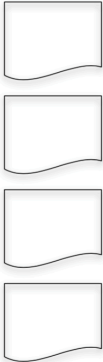
SCIENCE • VOL. 272 • 24 MAY 1996

Topic proportions and assignments

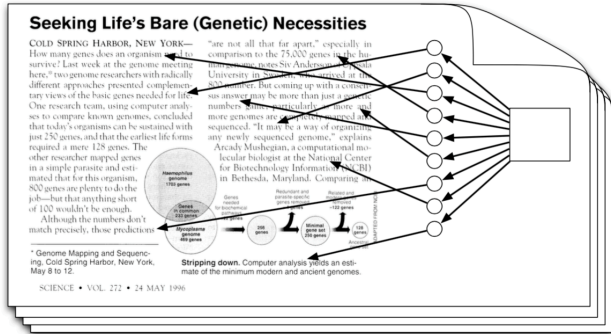


Latent Dirichlet Allocation (LDA)

Topics



Documents

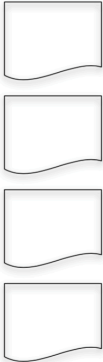


Topic proportions and assignments

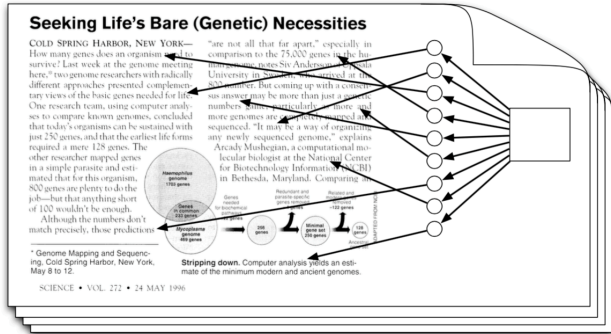
- In reality, we only observe the documents
- The other structure are **hidden variables**

Latent Dirichlet Allocation (LDA)

Topics



Documents



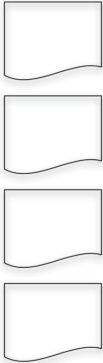
Topic proportions and assignments

- Our goal is to **infer** the hidden variables
- I.e., compute their distribution conditioned on the documents

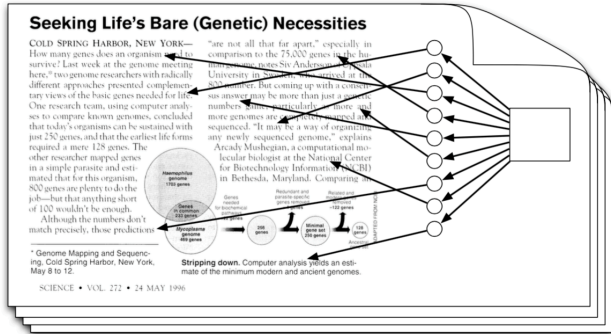
$$p(\text{topics, proportions, assignments} | \text{documents})$$

Latent Dirichlet Allocation (LDA)

Topics



Documents



Topic proportions and assignments

- Topic modelling allows us to extrapolate backwards from a collection of documents to infer the “topics” that could have generated them.

Latent Dirichlet Allocation (LDA)

- The LDA model is a Bayesian mixture model for discrete data where topics are assumed to be uncorrelated
- 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 over the parameters of the probability model from a combination of what is observed (words in documents) and what is hidden (topic and word parameters)

Latent Dirichlet Allocation: Details

- For each document, the LDA generative process is:
 1. randomly choose a distribution over topics (multinomial of length K)
 2. for each word in the document
 - 2.1 Probabilistically draw one of the K topics from the distribution over topics obtained in step 1, say topic k (each document contains topics in different proportions)
 - 2.2 Probabilistically draw one of the V words from β_k (each individual word in the document is drawn from one of the K topics in proportion to the document's distribution over topics as determined in previous step)
- The goal of inference in LDA is to discover the topics from the collection of documents, and to estimate the relationship of words to these, *assuming this generative process*

LDA generative model

1. Term distribution β for each topic is drawn:

$$\beta_k \sim \text{Dirichlet}(\eta)$$

→ probability that each word occurs in a given topic (k)

2. proportions θ of the topic distribution for the document are drawn by

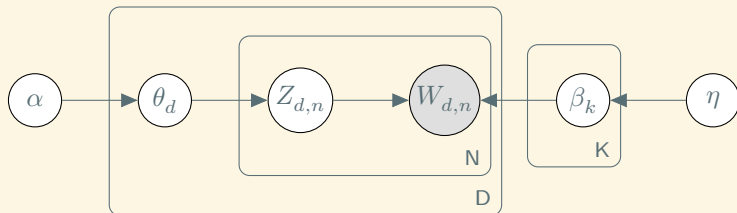
$$\theta_d \sim \text{Dirichlet}(\alpha)$$

→ probability that each topic occurs in a given document (d)

3. For each of the N words in each document

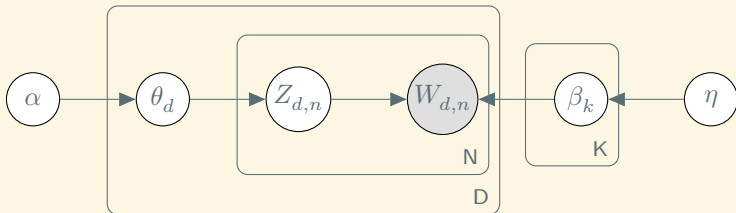
- choose a topic $z_i \sim \text{Multinomial}(\theta)$
- choose a word $w_i \sim \text{Multinomial}(p(w_i|z_i, \beta))$

LDA as a graphical model



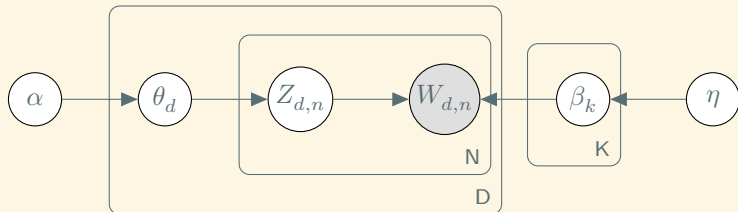
- Encodes **assumptions**
- Connects to **algorithms** for computing with data

LDA as a graphical model



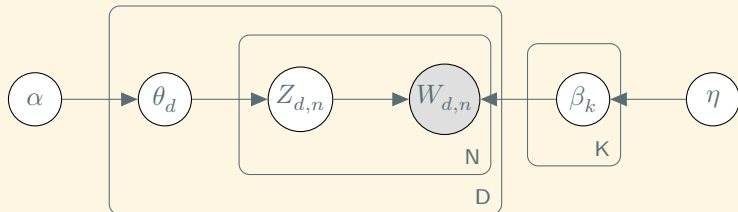
- Nodes are random variables; edges indicate dependence.
- Shaded nodes are observed; unshaded nodes are hidden.
- Plates indicate replicated variables.

LDA as a graphical model



- $W_{d,n}$ observed word (word level)
- $Z_{d,n}$ topic assignment (word level)
- θ_d topic proportions (document level)
- β_k probability distribution over words (topic level)
- α proportions parameter (corpus level)
- η topic parameter (corpus level)

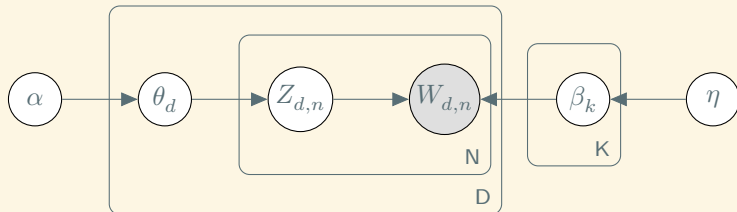
LDA as a graphical model



- $\beta_k \sim \text{Dirichlet}(\eta)$
- $\theta_d \sim \text{Dirichlet}(\alpha)$
- $Z_{d,n} \sim \text{Multinomial}(\theta_d)$
- $W_{d,n} \sim \text{Multinomial}(p(w_i|z_i, \beta_k))$

Note: η and α govern the *sparsity* of the draws from the dirichlet. As they $\rightarrow 0$, the multinomials become more sparse.

LDA as a graphical model

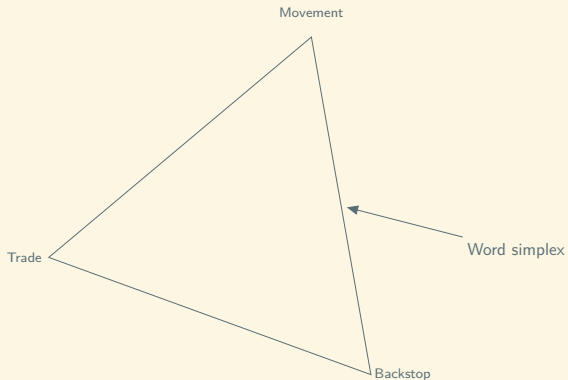


- This joint defines a posterior, $p(\theta, z, \beta|w)$.
- From a collection of documents, infer
 - Per-word topic assignment $z_{d,n}$
 - Per-document topic proportions θ_d
 - Per-corpus topic distributions β_k
- Then use posterior distribution over these parameters to perform the task at hand \rightarrow information retrieval, document similarity, exploration, and others.

The Dirichlet distribution

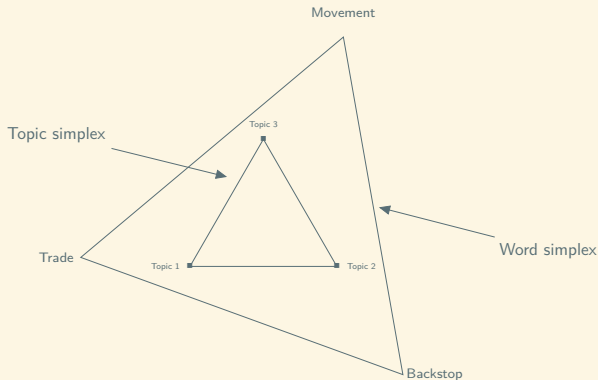
- The Dirichlet distribution is an exponential family distribution over the simplex, i.e., positive vectors that sum to one
- The parameter α controls the mean shape and sparsity of θ .
- The Dirichlet is used twice in LDA:
 - The topic proportions (θ) are a K dimensional Dirichlet
 - The topics (β) are a V dimensional Dirichlet.
- Estimation is performed using collapsed Gibbs sampling and/or Variational Expectation-Maximization (VEM)
- Fortunately, for us these are easily implemented in R

Latent Dirichlet allocation (LDA)



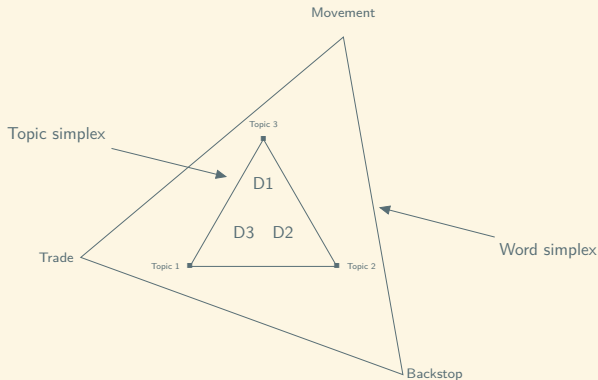
- Imagine a corpus consisting of only three words
- The word simplex describes the possible probabilities of the multinomial distribution over these three words

Latent Dirichlet allocation (LDA)



- We can locate **topics** within the **word-simplex**
- Each topic represents a different distribution over words
- Smaller $\eta \rightarrow$ sparser topics \rightarrow topics will be closer to the word-simplex lines

Latent Dirichlet allocation (LDA)

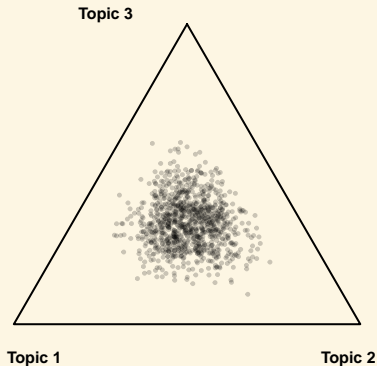


- We can locate **documents** within the **topic-simplex**
- Each document is a mixture of topics
- Smaller $\alpha \rightarrow$ sparser documents \rightarrow documents will be closer to topic-simplex lines

Dirichlet distribution

Recall that $\theta_d \sim \text{Dirichlet}(\alpha)$: the topic proportions of each document are governed by a dirichlet distribution with paramter α .

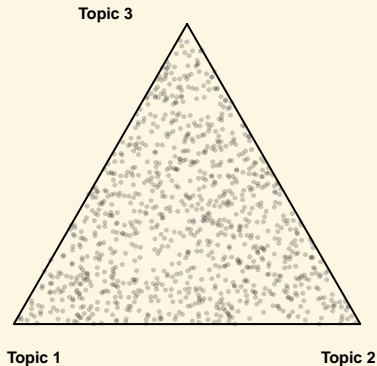
When $\alpha = 10$



Dirichlet distribution

Recall that $\theta_d \sim \text{Dirichlet}(\alpha)$: the topic proportions of each document are governed by a dirichlet distribution with paramter α .

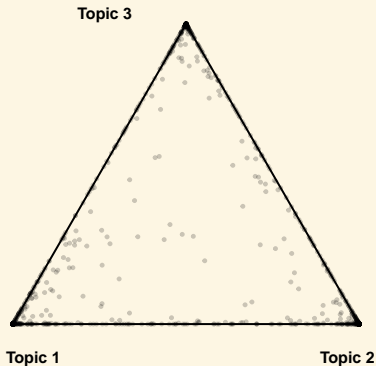
When $\alpha = 1$



Dirichlet distribution

Recall that $\theta_d \sim \text{Dirichlet}(\alpha)$: the topic proportions of each document are governed by a dirichlet distribution with paramter α .

When $\alpha = .1$



Why does LDA “work”?

- LDA trades off two goals.
 1. For each document, allocate its words to as few topics as possible.
(α)
 2. For each topic, assign high probability to as few terms as possible.
(η)
- These goals are at odds.
 1. Putting a document in a single topic makes (2) hard: All of its words must have probability under that topic.
 2. Putting very few words in each topic makes (1) hard: To cover a document's words, it must assign many topics to it.
- Trading off these goals finds groups of tightly co-occurring words.

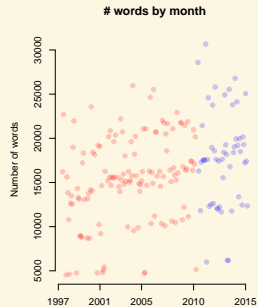
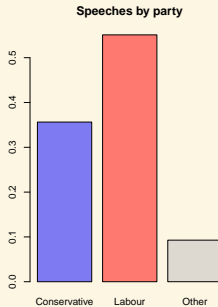
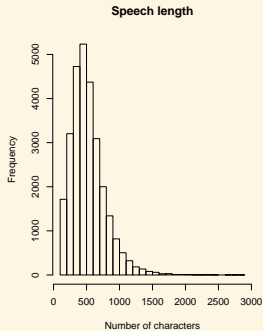
LDA summary

- LDA is a probabilistic model of text. It casts the problem of discovering themes in large document collections as a posterior inference problem.
- It lets us visualize the hidden thematic structure in large collections, and generalize new data to fit into that structure.
- LDA is a simple building block that enables many applications.
- It is popular because organizing and finding patterns in data has become important in the sciences, humanities, industry, and culture.
- Further, algorithmic improvements let us fit models to massive data.

LDA example

- Data: UK House of Commons' debates (PMQs)
 - ≈ 30000 parliamentary speeches from 1997 to 2015
 - ≈ 3000 unique words
 - $\approx 2m$ total words
- Note that I have already made a number of sample selection decisions
 - Only PMQs ($\approx 3\%$ of total speeches)
 - Removed frequently occurring & very rare words
 - All words have been stemmed
- Estimate a range of topic models ($K \in \{20, 30, \dots, 100\}$) using the `topicmodels` package

LDA example



Implementation in R (via quanteda)

```
library(quanteda)

## Create corpus
speechCorpus <- corpus(pmq, text_field = "Speech")

## Create and trim DFM
speechDFM <- dfm(speechCorpus,
                  remove = stopwords("en"), stem = T)
speechDFM <- dfm_trim(speechDFM, min_termfreq = 5)

## Convert for usage in 'topicmodels' package
tmDFM <- convert(speechDFM, to = 'topicmodels')

## Estimate LDA
ldaOut <- LDA(tmDFM, k = 60)

save(ldaOut, file = "ldaOut_60.Rdata")
```

LDA example

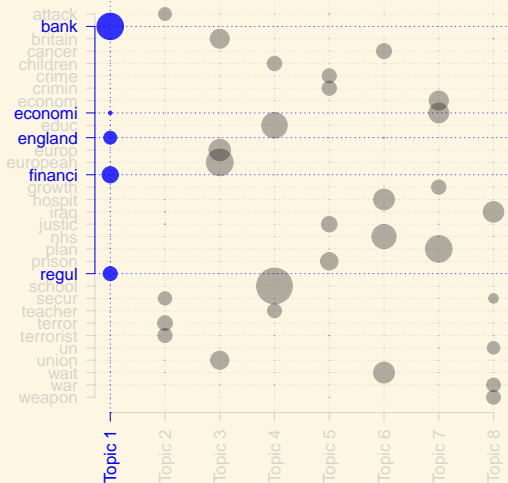
We will make use of the following score to visualise the posterior topics:

$$\text{term-score}_{k,v} = \hat{\beta}_{k,v} \log \left(\frac{\hat{\beta}_{k,v}}{(\prod_{j=1}^K \hat{\beta}_{j,v})^{\frac{1}{K}}} \right)$$

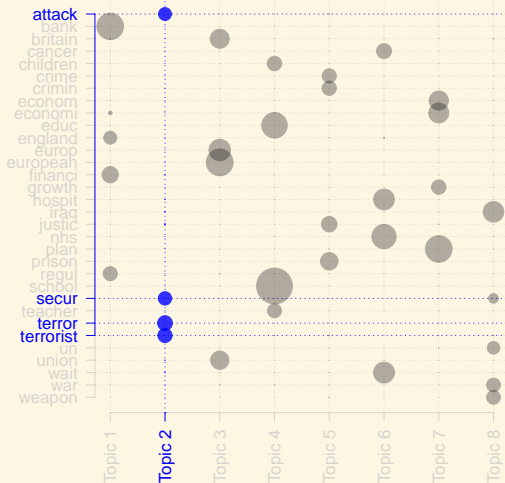
This formulation is similar to the TFIDF term score, where

- the first term, $\hat{\beta}_{k,v}$, is the probability of term v in topic k and is akin to the term frequency
- the second term is akin to the document frequency (i.e. it down-weights terms that have high probability under all topics)

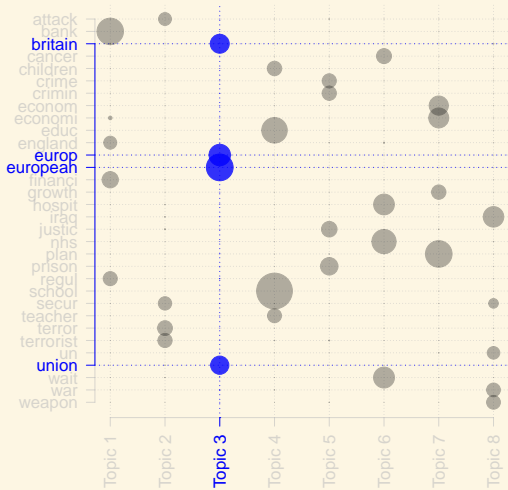
LDA example



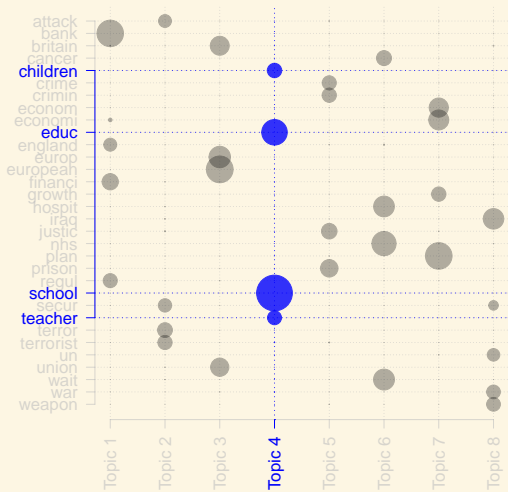
LDA example



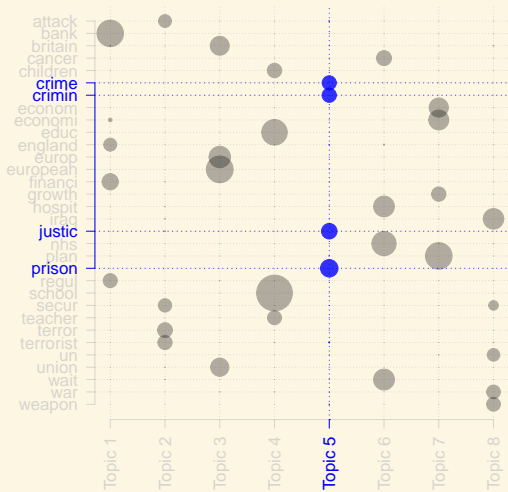
LDA example



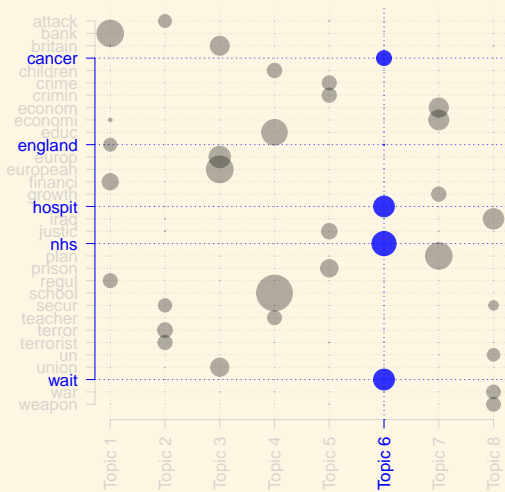
LDA example



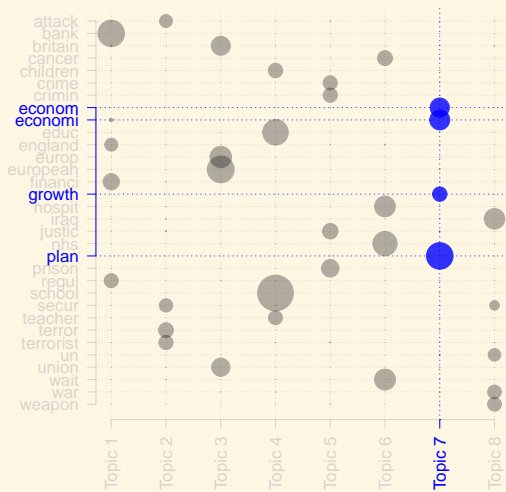
LDA example



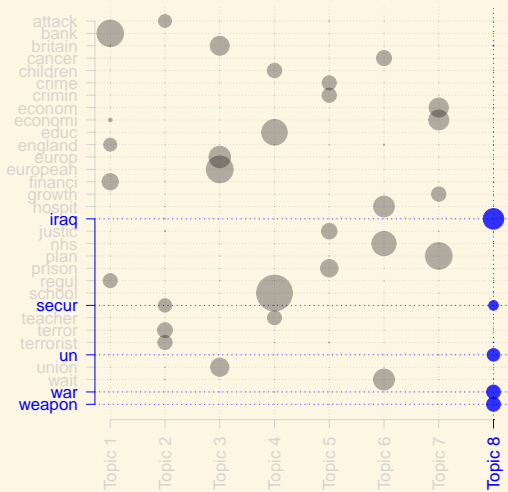
LDA example



LDA example



LDA example



LDA example

Topic 1

bank
financi
regul
england
crisi
fiscal
market

Topic 2

terror
terrorist
secur
attack
protect
agre
act

Topic 3

european
europ
britain
union
british
referendum
constitut

Topic 4

school
educ
children
teacher
pupil
class
parent

Topic 5

prison
justic
crimin
crime
releas
court
sentenc

Topic 6

nhs
wait
hospit
cancer
patient
list
health

Topic 7

plan
economi
econom
growth
grow
longterm
deliv

Topic 8

iraq
weapon
war
un
resolut
iraqi
saddam

LDA in research

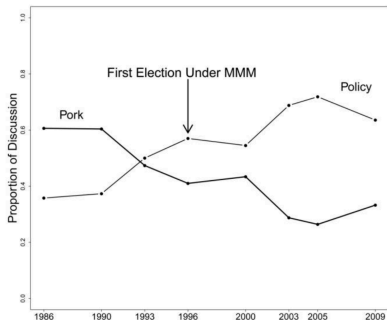
Research question: Do different electoral systems create incentives for politicians to focus on different aspects of policy?

Catalinac (2014) argues that the electoral reform in 1994 in Japan should increase the amount of attention that politicians devote to “policy” rather than “pork”.

LDA in research

Research question: Do different electoral systems create incentives for politicians to focus on different aspects of policy?

Catalinac (2014) argues that the electoral reform in 1994 in Japan should increase the amount of attention that politicians devote to “policy” rather than “pork”.

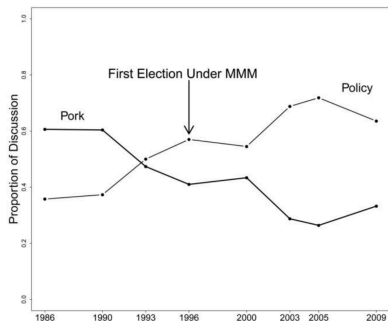


“Applying probabilistic topic modeling... shows that candidates for office change tried-and-true electoral strategies when confronted with an electoral reform.”

LDA in research

Research question: Do different electoral systems create incentives for politicians to focus on different aspects of policy?

Catalinac (2014) argues that the electoral reform in 1994 in Japan should increase the amount of attention that politicians devote to “policy” rather than “pork”.



“Applying probabilistic topic modeling... shows that candidates for office change tried-and-true electoral strategies when confronted with an electoral reform.” Questions:

- LDA on 8000 manifestos
- 69 topics (somewhat arbitrary)
- Dictionary? Supervised?

Break

Evaluating LDA performance

How can we tell how well a given topic model is performing?

Statistical approach:

- How well does the model predict held-out data?
- Ask which words the model believes will be in a given document and comparing this to the document's actual word composition (normally by splitting texts in half)
- Problems:
 - Prediction is not always important in exploratory or descriptive tasks. We may want models that capture other aspects of the data.
 - There tends to be a negative correlation between quantitative diagnostics such as these and human judgements of topic coherence!

Evaluating model performance

Human approach:

- Do topics have (human-identifiable) *semantic coherence*?
 - *Word intrusion*: ask subjects to identify a spurious word inserted into a topic
- Does the association between documents and topics make sense?
 - *Topic intrusion*: ask subjects to identify a topic that was not associated with the document by the model

Finding:

“Topic models which perform better on held-out likelihood may infer less semantically meaningful topics.” (Chang et al. 2009.)

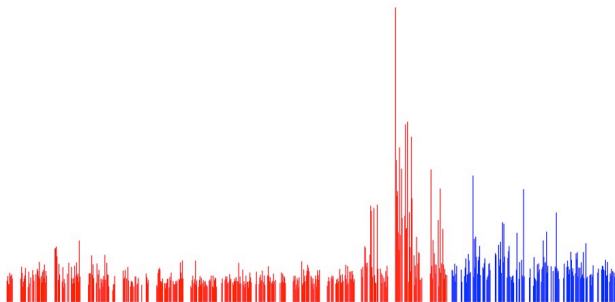
Evaluating LDA performance

- *Semantic validity* – does a topic identify a coherent groups of texts that are internally homogenous but distinctive from other topics?
- *Predictive validity* – how well does variation in topic usage correspond to known events?
- *Construct validity* – how well does our measure correlate with other measures?
- Here, we will focus on semantic and predictive validity. (Why?)

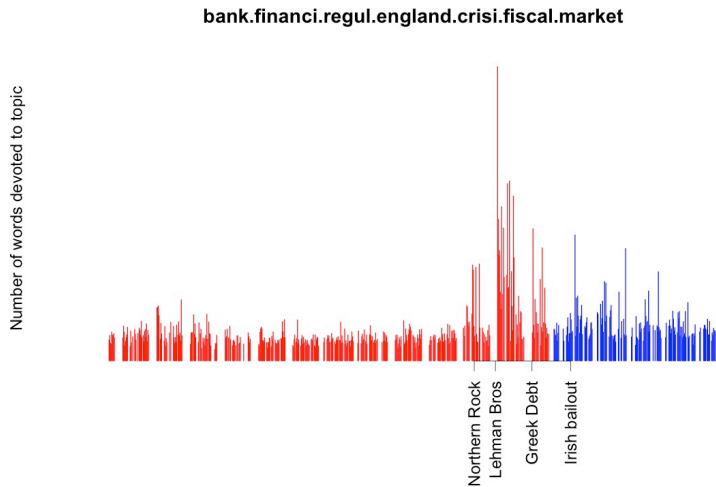
Predictive validity

bank.financi.regul.england.crisi.fiscal.market

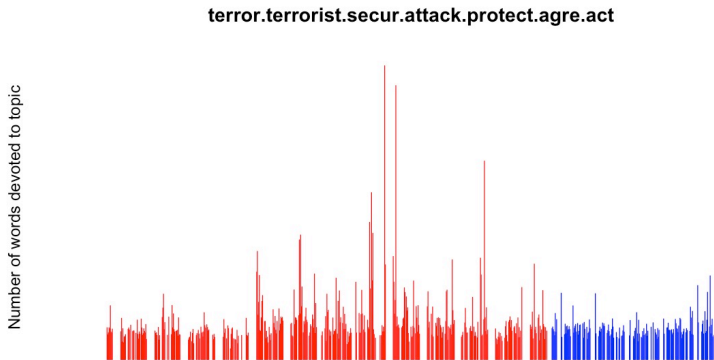
Number of words devoted to topic



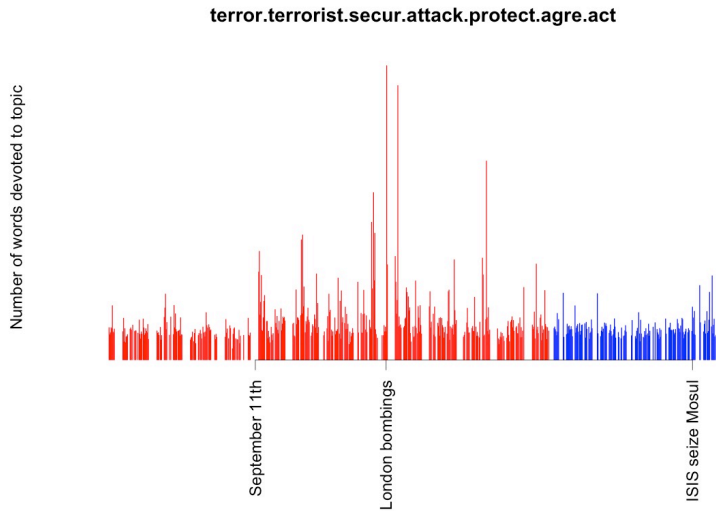
Predictive validity



Predictive validity



Predictive validity



Semantic validity

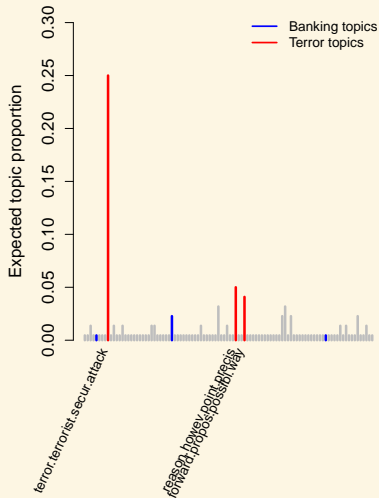
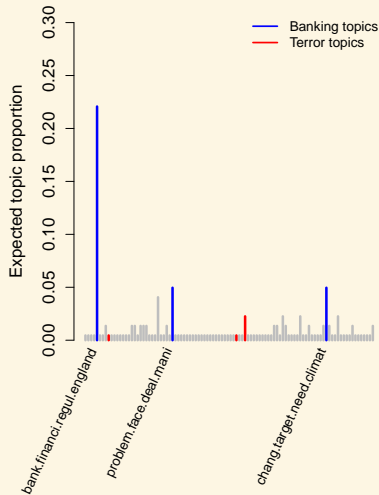
Consider the following texts:

The reforms that we are bringing into the banking system will include greater competition in banking. We will have a judgment from the European Commission soon, which we are supporting, that will allow more competition in British banking. As for the restructuring of the banking system and whether there should be investment banks on one side and retail-only banks on the other, the right hon. Gentleman must remember that Northern Rock was effectively a retail bank and it collapsed. Lehman Brothers was effectively an investment bank without a retail bank and it collapsed. The difference between retail and investment banks is not the cause of the problem. The cause of the problem is that banks have been insufficiently regulated at a global level and we have to set the standards for that for the future. We will be doing that at the G20 Finance Ministers summit in a few weeks' time.

The purpose of this coming before the House is for the Home Secretary to advise us that, in her view, there is an exceptional terrorist threat a grave terrorist threat that either has occurred or is occurring and that the need for action is urgent, but that it has not been possible to assemble the necessary evidence to lay charges within the 28 days. It will then be for the House to vote on the commencement order and agree that an exceptional terrorist incident has occurred. It is not the business of the House to interfere in the individual case, but it should be able to vote simply on whether an exceptional and grave terrorist threat has occurred. Given that the right hon. Gentleman and others have referred to the Civil Contingencies Act 2004 in discussing this issue, I would hope that he understands that this is exactly the same problem that has to be faced in respect of that Act.

We will call these the banking and terrorism texts.

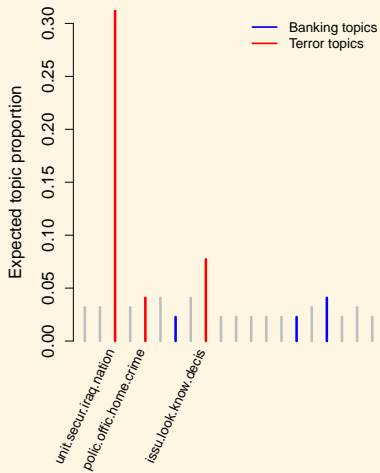
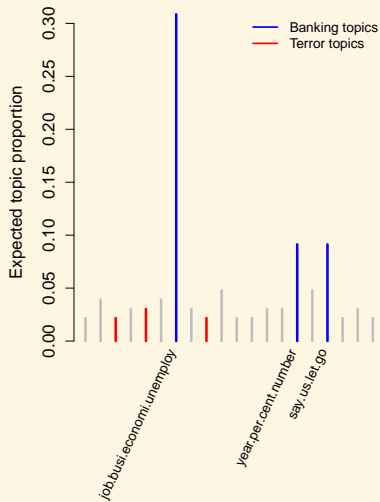
Semantic validity



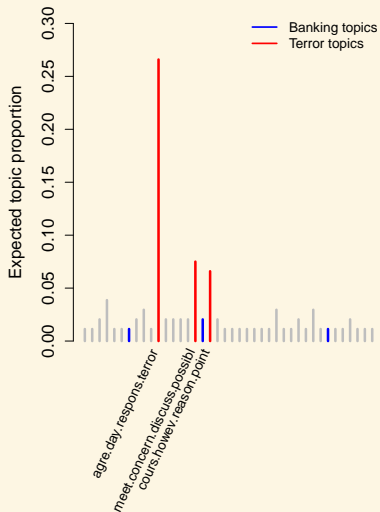
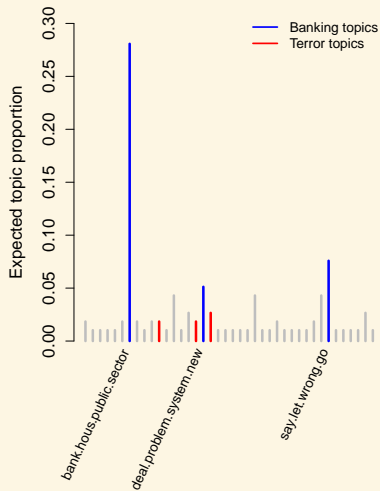
Semantic validity

- These plots suggest that our model is picking up at least some properties that we would intuitively expect to see in this particular corpus
- However, they do not help us to choose between the different models that we have estimated
- In other words, how should we pick K ?

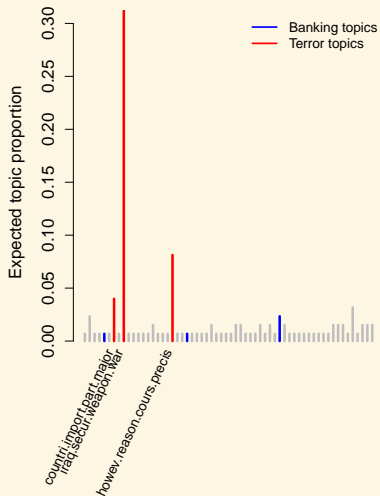
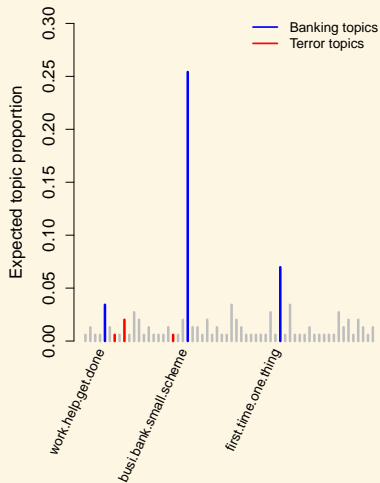
Which K?



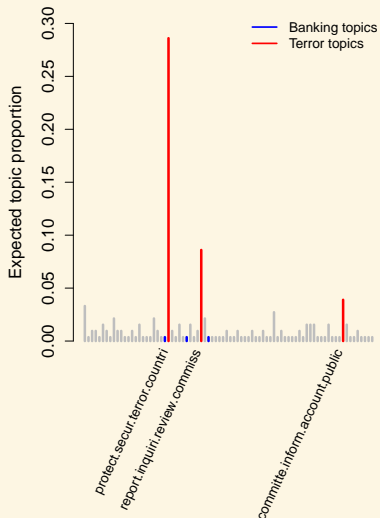
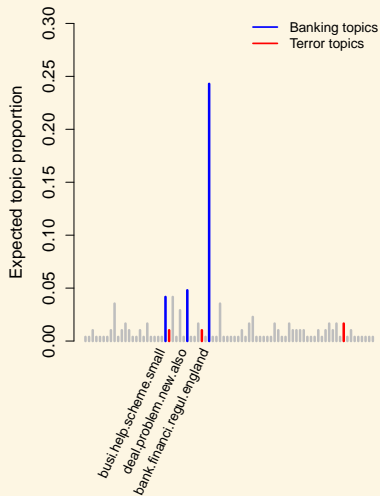
Which K?



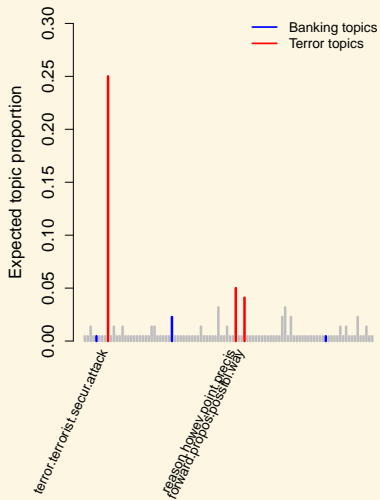
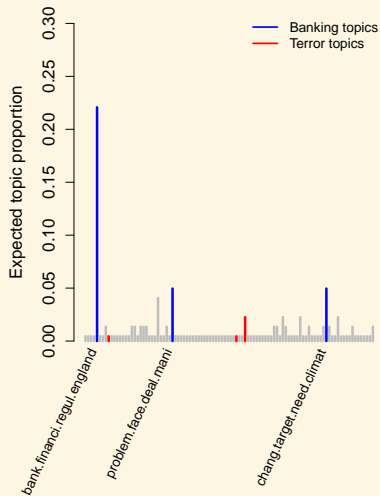
Which K?



Which K?



Which K?



Semantic validity (revisited)

- Semantic validity requires that topics are coherent and meaningful.
- We hope that texts assigned to a given topic are homogenous
- We hope that texts from different topics are distinctive
- A strong test, but requires human input, which can be costly
- One option would be to crowdsource validation using online workers
 - Benoit et. al (2015) *Crowd-Sourced Text Analysis: Reproducible and Agile Production of Political Data*
- Another option is to mercilessly exploit a class of students (not you, don't worry)

Semantic validity (revisited)

Your task is simply to read these short texts, and use the answer box to tell me whether the pair of speeches you are looking at are 'unrelated', 'loosely related' or 'closely related' in terms of the topics under consideration.

Related?

Unrelated ▼

NEXT COMPARISON.

Comparisons completed: 0

Text one:

That is total complacency about one month's figures when the Prime Minister has had five years of failure under this Government. Under this Prime Minister we are a country of food banks and bank bonuses; a country of tax cuts for millionaires while millions are paying more. Is not his biggest broken promise of all that we are all in it together?

Text two:

This is totally desperate stuff because the Prime Minister has nothing to say about the cost of living crisis. That is the reality, and his reshuffle had nothing to do with the country and everything to do with his party. After four years of this Government, we have a recovery that people cannot feel, a cost of living crisis that people cannot deny, and a Prime Minister whom people cannot believe.

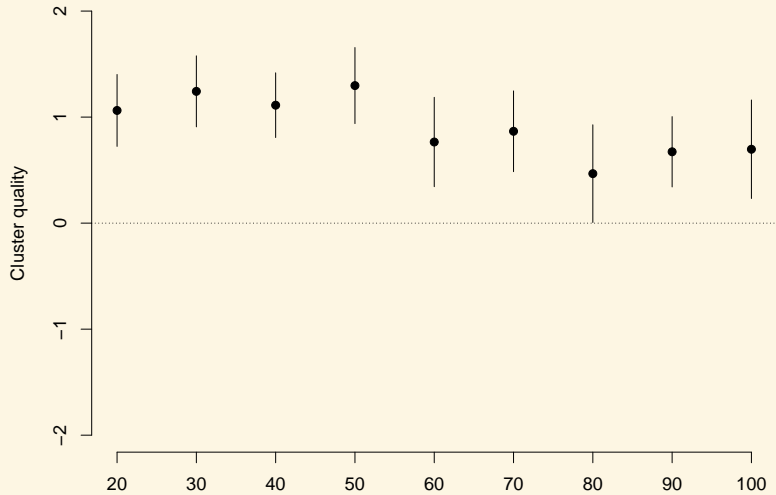
Semantic validity (revisited)

- Sample pairs of speeches from the posterior distribution
 - 5 pairs from the same topic, for each topic
 - 5 pairs from different topics, for each topic
- Randomly present to human coders, asking whether they are:
 - closely related (3); loosely related (2); unrelated (1)
- Calculate the Cluster Quality for the topic by regressing

$$Related_{ik} = \alpha + \beta_k * SameTopic_{ik}$$

- β_k is an estimate of the cluster quality of topic model k
 - i.e. the difference between relatedness of same-topic and different-topic pairs
- Repeat for each value of K

Semantic validity (revisited)



An application

- Once we are happy with the topic model we have estimated, we can use the posterior distribution in various ways
 - Visualisation
 - Information retrieval
 - Corpus exploration
 - Similarity
 - Dimensionality reduction
- In this example, we can use the posterior distribution of document-topic proportions to ask: Which MPs are most active at asking questions in each topic?

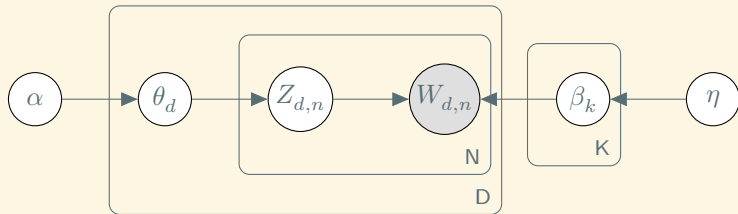
$$MPAttention_{i,k} = \frac{MPWords_{i,k}}{\sum_1^K MPWords_{i,k}}$$

An application

bank.financi.regul	terror.terrorist.secur	european.europ.britain	school.educ.children
Christopher Gill	Shahid Malik	David Lock	Christine Butler
Malcolm Wicks	Parmjit Gill	William Cash	Melanie Johnson
Tom Greatrex	George Mudie	Alistair Darling	Julie Kirkbride
Alasdair Morgan	Jonathan Djanogly	Denzil Davies	Sam Gyimah
Nick Herbert	James Brokenshire	David Heathcoat-Amory	Malcolm Moss
Karl McCartney	Tobias Ellwood	David Wilshire	Paul Clark
Donald Gorrie	Brian Wilson	David Davis	Ian Liddell-Grainger
Justin Tomlinson	John Maples	Giles Radice	Michael Heseltine
John Townend	Seamus Mallon	Ann Winterton	Stephen Hammond
Howard Flight	Ann Keen	Jenny Jones	Chris Pond
Derek Foster	Stephen Barclay	Dale Campbell-Savours	Ivan Henderson
Lindsay Roy	Pat McFadden	Jacob Rees-Mogg	Derek Conway
prison.justic.crimin	nhs.wait.hospit	plan.economi.econom	iraq.weapon.war
Jack Lopresti	Julia Goldsworthy	Chloe Smith	Alan Howarth
Kevin McNamara	Seema Malhotra	Conor Burns	Chris Smith
Alan Clark	Michael Penning	Donald Gorrie	Tony Worthington
Chris Skidmore	Nick Hurd	Guy Opperman	Terry Davis
Charles Walker	Virendra Sharma	Karen Bradley	George Foulkes
Jeremy Wright	Tim Farron	Neil Carmichael	Jonathan Sayeed
Tess Kingham	Bill Esterson	Wayne David	Melanie Johnson
Sarah Champion	John Penrose	Michael Colvin	Denzil Davies
Philip Davies	Malcolm Chisholm	Michael Ellis	Paul Stinchcombe
Kali Mountford	Grant Shapps	Anne Milton	Adam Price
Mike Wood	Marion Roe	John Stevenson	Kevin Hughes
Lynda Waltho	Mike Thornton	Sarah Newton	Tony Benn

Beyond Latent Dirichlet Allocation

LDA summary



- LDA is a simple topic model.
- It can be used to find topics that describe a corpus.
- Each document exhibits multiple topics.
- There are several ways to extend this model.

Extending LDA

- LDA can be **embedded in more complicate models**, embodying further intuitions about the structure of the texts.
- E.g., it can be used in models that account for syntax, authorship, word sense, dynamics, correlation, hierarchies, and other structure.
- The **data generating distribution** can be changed. We can apply mixed-membership assumptions to many kinds of data.
- E.g., we can build models of images, social networks, music, purchase histories, computer code, genetic data, and other types.
- The **posterior** can be used in creative ways.
- E.g., we can use inferences in information retrieval, recommendation, similarity, visualization, summarization, and other applications.

Extending LDA

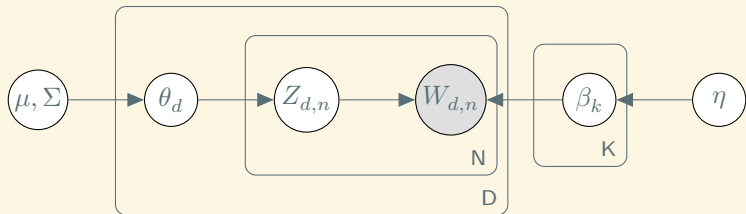
- These different kinds of extensions can be combined.
- To give a sense of how LDA can be extended, we'll look at several examples of major extensions.
- We will discuss
 - Correlated topic models
 - Dynamic topic models
 - Structural topic models

Correlated and Dynamic Topic Models

Correlated topic model

- The Dirichlet is a distribution on the simplex, positive vectors that sum to 1.
- It assumes that components are nearly independent.
- In real data, an article about fossil fuels is more likely to also be about geology than about genetics.
- The logistic normal is a distribution on the simplex that can model dependence between components (Aitchison, 1980).
- Re-parameterise so that the (log of the) parameters of the topic-proportions multinomial are drawn from a multivariate Gaussian distribution

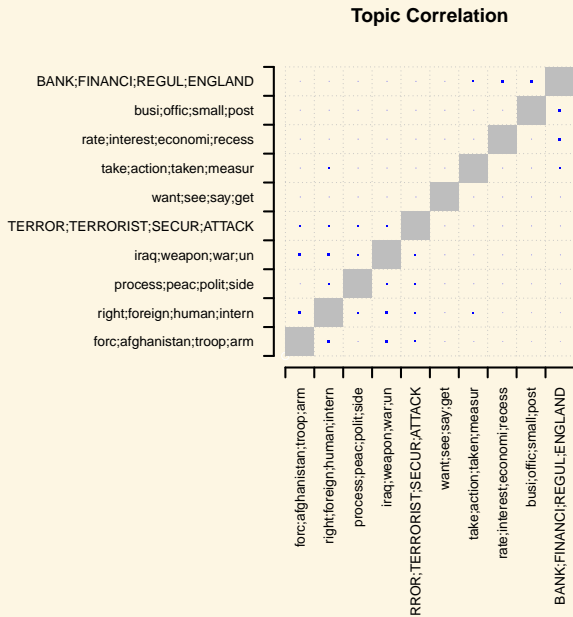
Correlated topic model



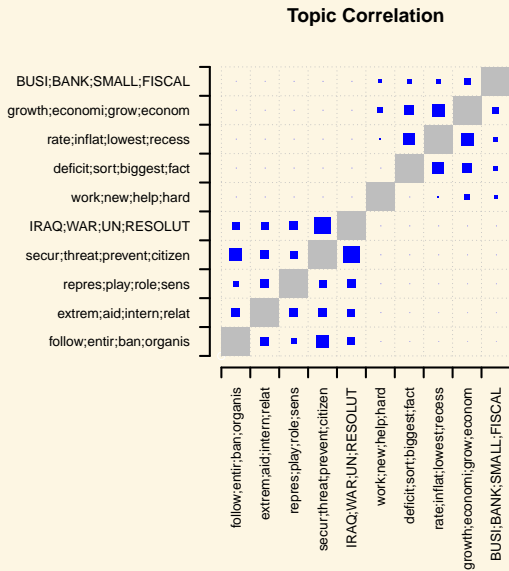
where the first node is logistic normal prior.

- Draw topic proportions from a logistic normal.
- This allows topic occurrences to exhibit correlation.
- Provides a “map” of topics and how they are related
- Provides a better fit to text data, but computation is more complex

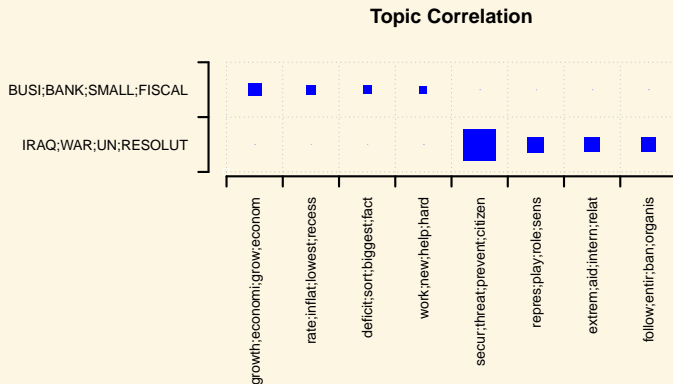
LDA topic correlation



CTM topic correlation



CTM topic correlation



Dynamic topic model

- LDA assumes that the order of documents does not matter.
- Not appropriate for sequential corpora (e.g., that span hundreds of years)
- We may want to track how language changes over time.
 - How has the language used to describe neuroscience developed from “The Brain of Professor Laborde” (1903) to “Reshaping the Cortical Motor Map by Unmasking Latent Intracortical Connections” (1991)
 - How has the language used to describe love developed from “Pride and Prejudice” (1813) to “Eat, Pray, Love” (2006)
- Dynamic topic models let the topics drift in a sequence.

Dynamic topic model

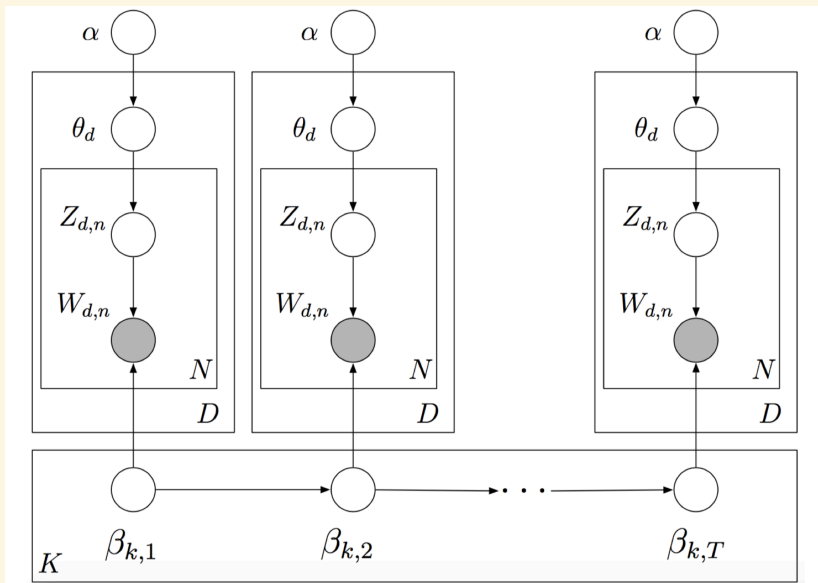


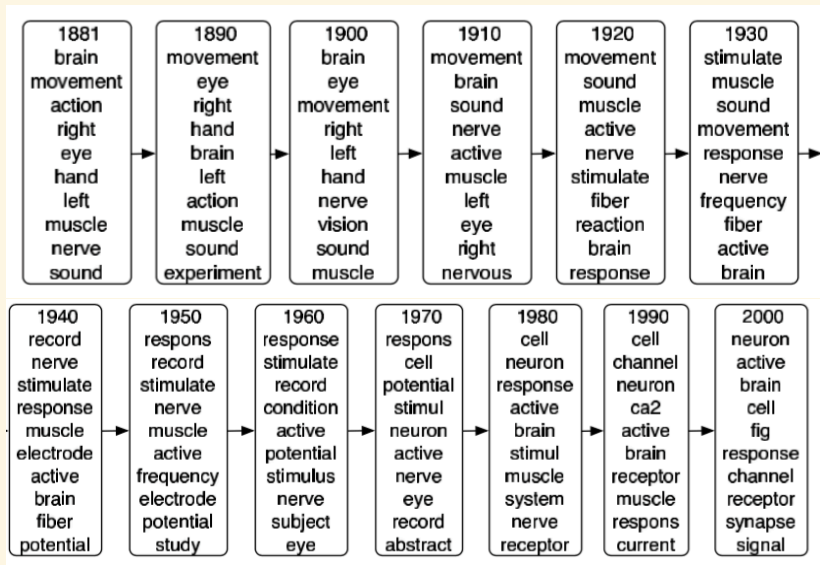
Plate (K) allows topics to “drift” through time.

Dynamic topic models



- Use a logistic normal distribution to model topics evolving over time.
 - The k th topic at time 2 has evolved smoothly from the k th topic at time 1
- As for CTMs, this makes computation more complex. But it lets us make inferences about sequences of documents.

Dynamic topic model (example)



Summary: Correlated and dynamic topic models

- The Dirichlet assumption on topics and topic proportions makes strong conditional independence assumptions about the data.
- The **correlated topic model** uses a logistic normal on the topic proportions to find patterns in how topics tend to co-occur.
- The **dynamic topic model** uses a logistic normal in a linear dynamic model to capture how topics change over time.
- What's the catch? These models are harder to compute.

Structural Topic Model

Structural topic model

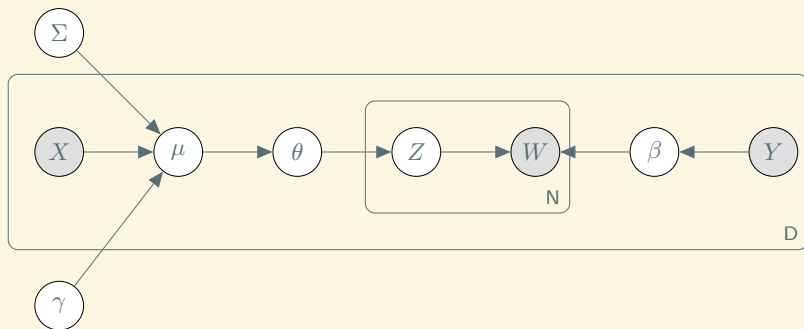
- Typically, when estimating topic models we are interested in how some covariate is associated with the prevalence of topic usage (Gender, date, political party, etc)
- The Structural Topic Model (STM) allows for the inclusion of arbitrary covariates of interest into the generative model
- The addition of covariates provides structure to the prior distributions
 1. Benefit 1: improves the estimation of the topics by allowing documents to share information according to the covariates (known as 'partial pooling' of parameters)
 2. Benefit 2: the relationship between covariates and latent topics is most frequently the estimand of interest, so we should include this in the estimation procedure

Structural topic model

How does it differ from LDA?

- As with the CTM, topics within the STM can be **correlated**
- **Topic prevalence** is allowed to vary according to the covariates X
 - Each document has its own prior distribution over topics, which is defined by its covariates, rather than sharing a global mean
- **Topical content** can also vary according to the covariates Y
 - Word use *within* a topic can differ for different groups of speakers/writers

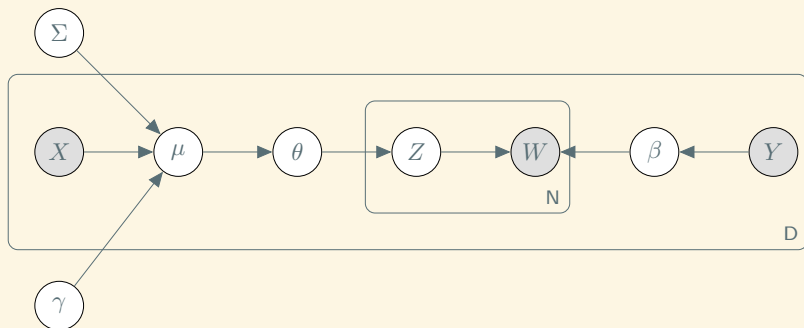
Structural topic model



Topic prevalence model:

- Draw topic proportions from a logistic normal generalised linear model based on covariates X
- This allows the expected document-topic proportions to vary by covariates, rather than from a single shared prior

Structural topic model



Topical content model:

- The β coefficients, which indicate the distribution over words for a given topic, are allowed to vary according to the covariates Y
- This allows us to estimate how different covariates affect the words used *within a given topic*

Structural Topic Model (example)

- In the legislative domain, we might be interested in the degree to which MPs from different parties represent distinct interests in their parliamentary questions
- We can use the STM to analyse how topic prevalence varies by party

```
## Set topic count and estimate STM
```

```
K <- 60
```

```
stmOut <- stm(  
  documents = speechDFM,  
  data= docvars(speechDFM),  
  prevalence = ~party,  
  content = ~party,  
  K = K,  
  seed = 123)
```

Structural Topic Model (example)

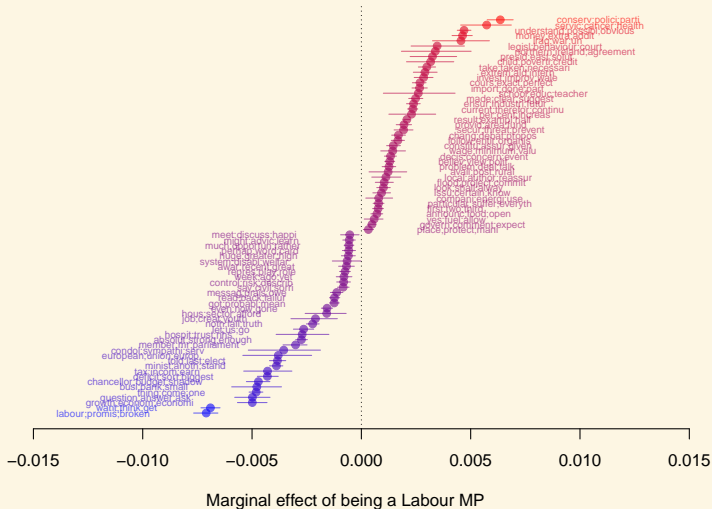
- Specify a linear model with:
 - the topic proportions of speech d , by legislator i as the outcome
 - the party of legislator i as the predictor

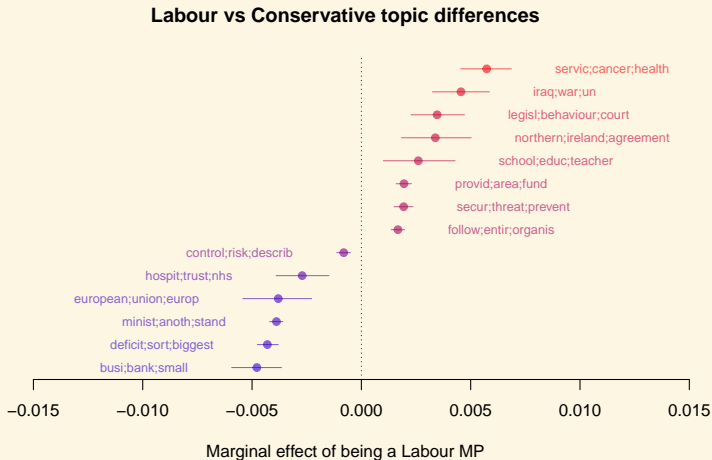
$$\theta_{dk} = \alpha + \gamma_{1k} * \text{labour}_{d(i)}$$

- The γ_k coefficients give the estimated difference in topic proportions for Labour and Conservative legislators for each topic

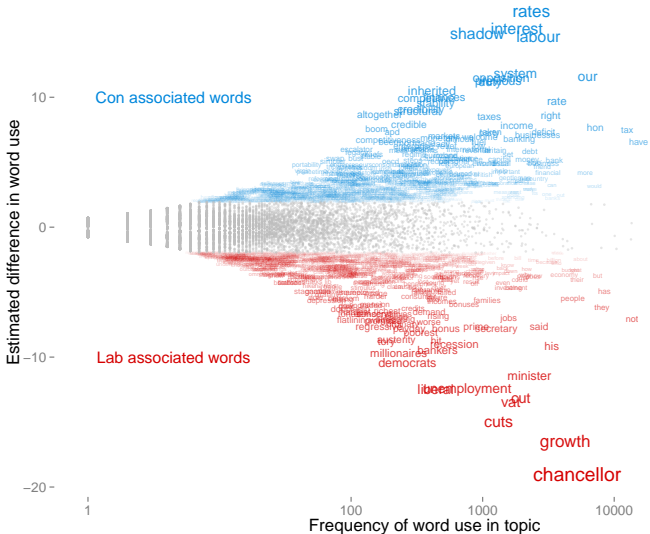
Topic prevalence

Labour vs Conservative topic differences

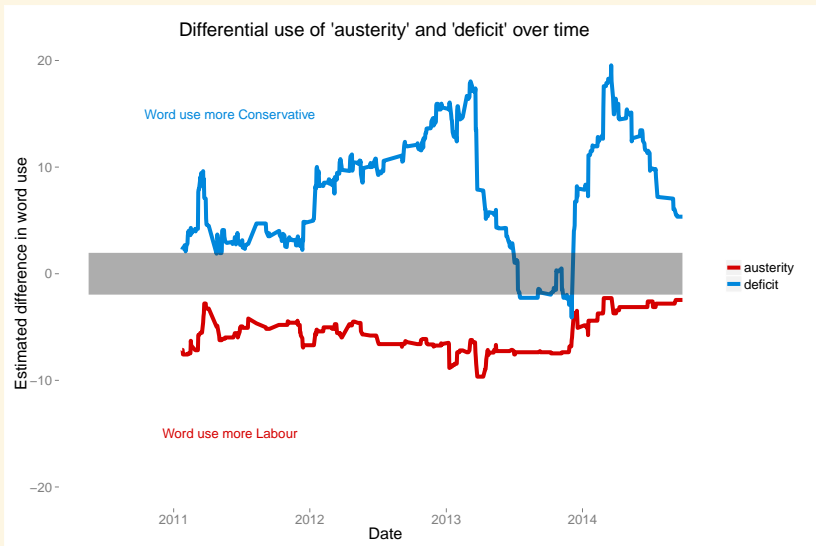




Topic content



Topic content



Summary

Topic Model Summary

Topic models assume:

- That there are K topics shared by a corpus collection.
- That each document exhibits the topics with different proportions.
- That each word is drawn from one topic.
- That we can *discover* the structure that best explain a corpus.

Topic models can be adapted to many settings

- Relax assumptions
- Combine models
- Model more complex data

Implementations of topic models in R

Incomplete list:

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