Topic Models

Pawan Goyal

CSE, IITKGP

October 8-9, 2015

Why Topic Modeling?

Information Overload

As more information becomes available, it becomes more difficult to find and discover what we need.

Why Topic Modeling?

Information Overload

As more information becomes available, it becomes more difficult to find and discover what we need.

Main Tools: Search and Links

- We type keywords into a search engine and find a set of related documents
- We look at these documents and possibly navigate to other documents

Why Topic Modeling?

Search Based-on themes

- Imagine searching and exploring documents based on themes that run through them.
- We might “zoom-in” or “zoom-out” to find specific or broader themes
- We might look at how themes change through time, how they are connected to each other
- Find the theme first and then examine the documents pertaining to that theme

Why Topic Modeling?

Topic Modeling

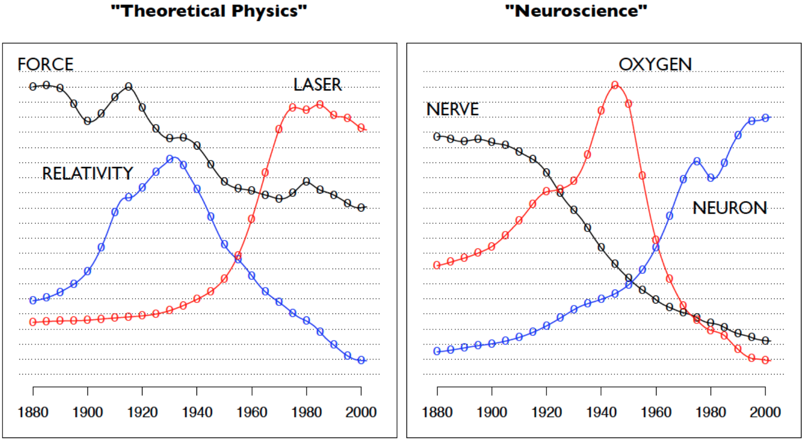
Provides methods for automatically organizing, understanding, searching and summarizing large electronic archives without any prior annotation or labeling

- Discover the hidden themes that pervade the collection
- Annotate the documents according to those themes
- Use annotations to organize, summarize, and search the texts

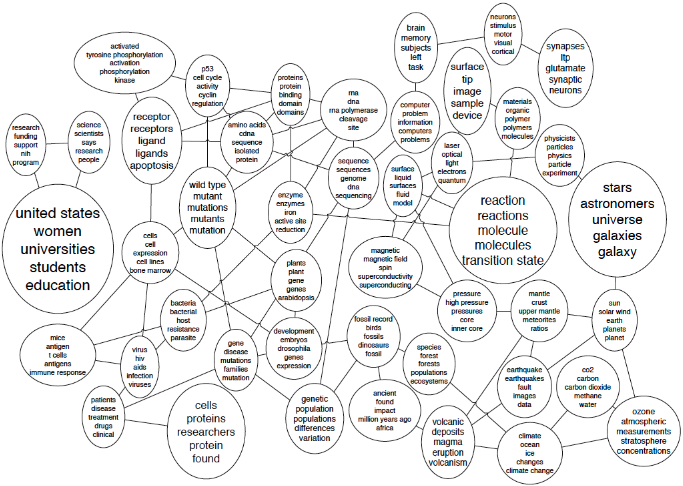
Applications: Discover Topics from a corpus

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

Applications: Model the evolution of topics over time



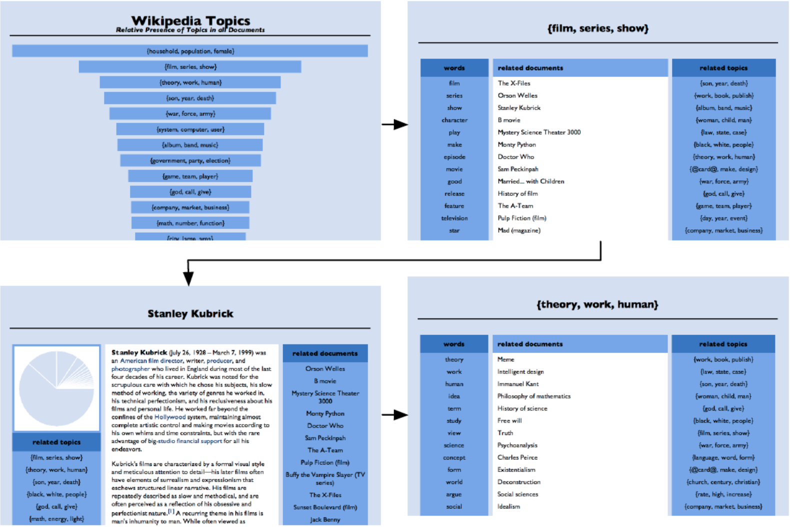
Applications: Model connections between topics



Link Prediction using Relational Topic Models

Markov chain Monte Carlo convergence diagnostics: A comparative review	
Minorization conditions and convergence rates for Markov chain Monte Carlo	RTM (ψ_e)
Rates of convergence of the Hastings and Metropolis algorithms	
Possible biases induced by MCMC convergence diagnostics	
Bounding convergence time of the Gibbs sampler in Bayesian image restoration	
Self regenerative Markov chain Monte Carlo	
Auxiliary variable methods for Markov chain Monte Carlo with applications	
Rate of Convergence of the Gibbs Sampler by Gaussian Approximation	
Diagnosing convergence of Markov chain Monte Carlo algorithms	
Exact Bound for the Convergence of Metropolis Chains	LDA + Regression
Self regenerative Markov chain Monte Carlo	
Minorization conditions and convergence rates for Markov chain Monte Carlo	
Gibbs-markov models	
Auxiliary variable methods for Markov chain Monte Carlo with applications	
Markov Chain Monte Carlo Model Determination for Hierarchical and Graphical Models	
Mediating instrumental variables	
A qualitative framework for probabilistic inference	
Adaptation for Self Regenerative MCMC	

Applications: Organize and browse large corpora

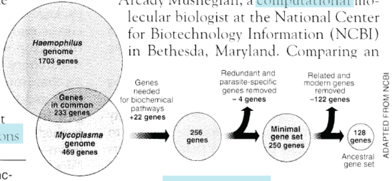


https://www.princeton.edu/ achaney/tmve/wiki100k/browse/topic-presence.html

Seeking Life's Bare (Genetic) Necessities

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

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



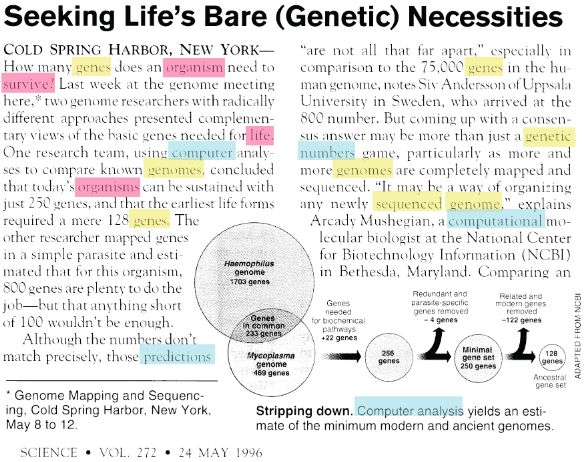
Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

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

SCIENCE • VOL. 272 • 24 MAY 1996

This articles is about using data analysis to determine the number of genes an organism needs to survive

Intuition: Documents exhibit multiple topics



Seeking Life's Bare (Genetic) Necessities

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

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

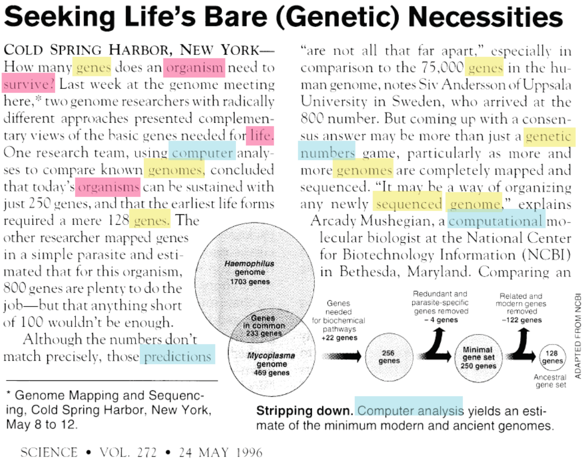
Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

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

SCIENCE • VOL. 272 • 24 MAY 1996

Highlighted words: 'blue': data analysis, 'pink': evolutionary biology, 'yellow': genetics

Intuition: Documents exhibit multiple topics



Seeking Life's Bare (Genetic) Necessities

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

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

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

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

SCIENCE • VOL. 272 • 24 MAY 1996

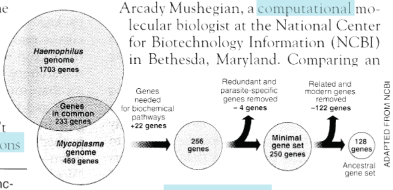
The article blends genetics, data analysis and evolutionary biology in different proportions

Intuition: Documents exhibit multiple topics

Seeking Life's Bare (Genetic) Necessities

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

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



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

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

SCIENCE • VOL. 272 • 24 MAY 1996

Knowing that this article blends those topics would help situate it in a collection of scientific articles

Topic Model: Basic Idea

A generative statistical model that captures this intuition.

Generative Model

Documents are mixture of topics, where a topic is a probability distribution over words.

Topic Model: Basic Idea

A generative statistical model that captures this intuition.

Generative Model

Documents are mixture of topics, where a topic is a probability distribution over words.

genetics topic has words about genetics with high probability and the *evolutionary biology* topic has words about evolutionary biology with high probability.

Topic Model: Basic Idea

A generative statistical model that captures this intuition.

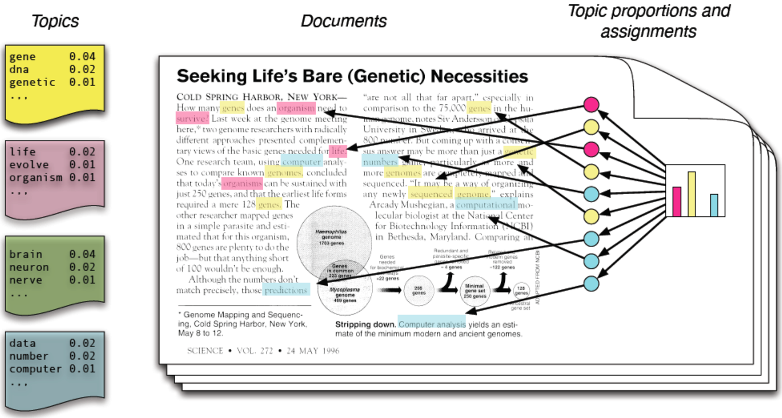
Generative Model

Documents are mixture of topics, where a topic is a probability distribution over words.

genetics topic has words about genetics with high probability and the *evolutionary biology* topic has words about evolutionary biology with high probability.

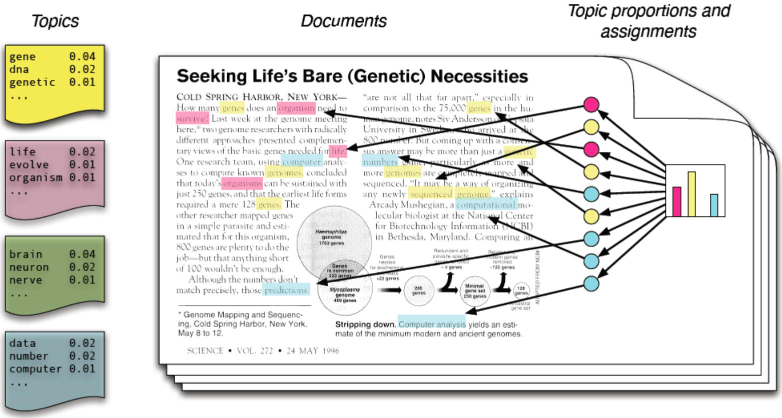
Technically, the generative model assumes that the topics are generated first, before the documents.

Generative Model for LDA



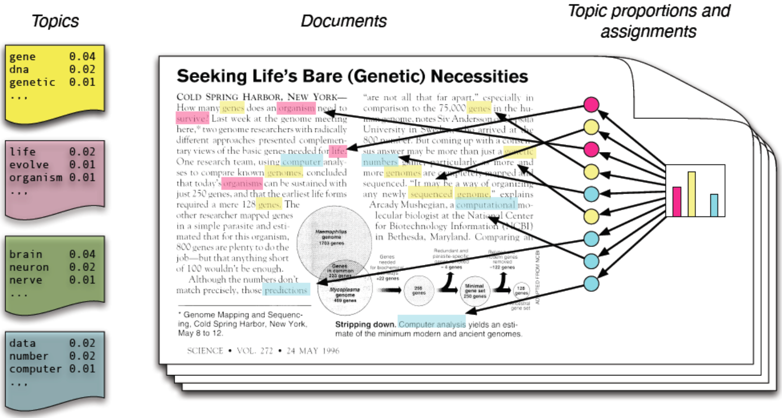
- Each **topic** is a distribution over words
- Each **document** is a mixture of corpus-wide topics
- Each **word** is drawn from one of those topics

Generative Model for LDA



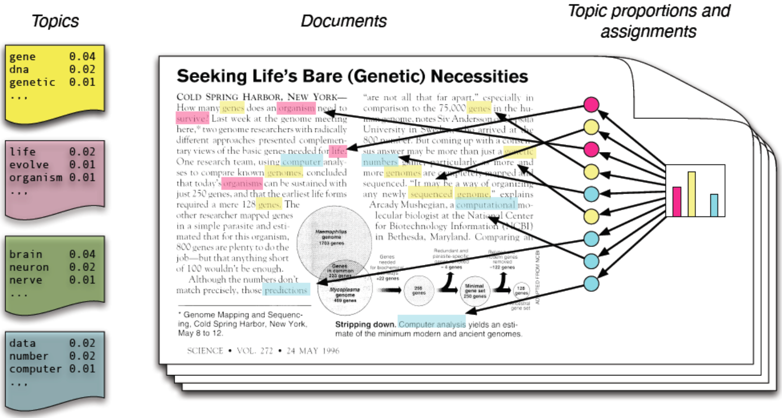
- Each **topic** is a distribution over words
- Each **document** is a mixture of corpus-wide topics
- Each **word** is drawn from one of those topics

Generative Model for LDA



- 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

Generative Model for LDA



- 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

What does the statistical model reflect?

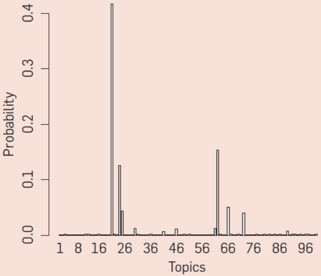
- All the document in the collection share the same set of topics, but each document exhibits those topics in different proportions
- Each word in each document is drawn from one of the topics, where the selected topic is chosen from the per-document distribution over topics

What does the statistical model reflect?

- All the document in the collection share the same set of topics, but each document exhibits those topics in different proportions
- Each word in each document is drawn from one of the topics, where the selected topic is chosen from the per-document distribution over topics

In the example article, the distribution over topics would place probability on *genetics*, *data analytics* and *evolutionary biology*, and each word is drawn from one of those three topics.

Real Inference with LDA for the example article

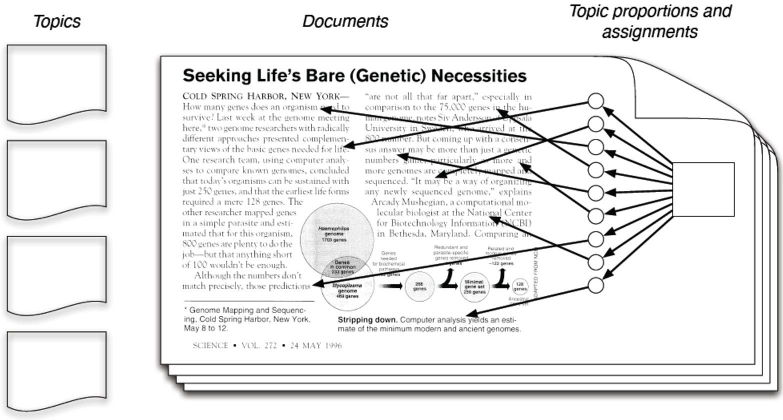


"Genetics"	"Evolution"	"Disease"	"Computers"
human	evolution	disease	computer
genome	evolutionary	host	models
dna	species	bacteria	information
genetic	organisms	diseases	data
genes	life	resistance	computers
sequence	origin	bacterial	system
gene	biology	new	network
molecular	groups	strains	systems
sequencing	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

Central Problem of LDA

- The documents themselves are observed, while the topic structure - the topics, per-document topic distributions, and the per-document per-word topic assignments - is *hidden structure*.
- The central computational problem is to use the observed documents to infer the hidden topic structure, i.e. *reversing* the generative process.

Goal: The posterior distribution



Infer the hidden variables

Compute their distribution conditioned on the documents

Topics from TASA corpus

37,000 text passages from educational materials (300 topics)

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

Topics from TASA corpus

Documents with different content can be generated by choosing different distributions over topics.

- Equal probability to first two topics:

Topics from TASA corpus

Documents with different content can be generated by choosing different distributions over topics.

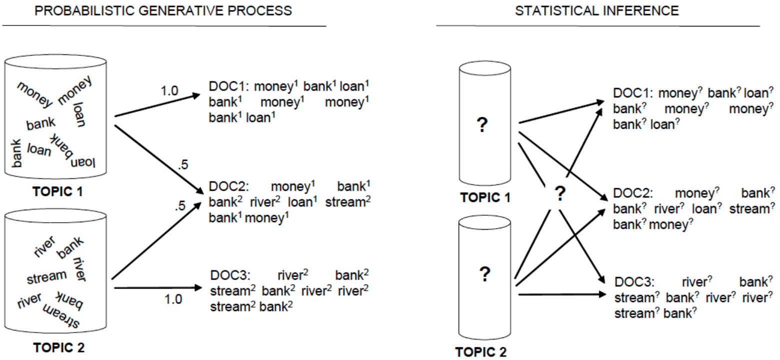
- Equal probability to first two topics: about a person who has taken too many drugs and how that affected color perceptions.
- Equal probability to the last two topics:

Topics from TASA corpus

Documents with different content can be generated by choosing different distributions over topics.

- Equal probability to first two topics: about a person who has taken too many drugs and how that affected color perceptions.
- Equal probability to the last two topics: about a person who experienced a loss of memory, which required a visit to the doctor.

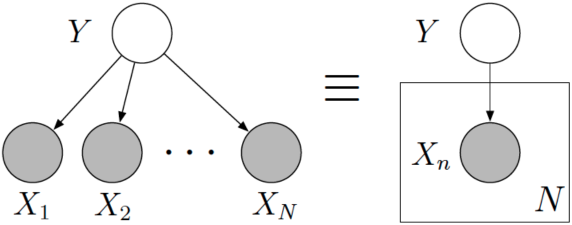
Generative model and statistical inference



Important points

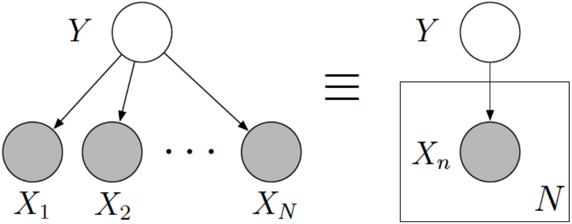
- *bag-of-words assumption:* The generative process does not make any assumptions about the order of words in the documents.
- *capturing polysemy:* The way that the model is defined, there is no notion of mutual exclusivity that restricts words to be part of one topic only. Ex: both ‘money’ and ‘river’ topics can give high probability to the word ‘bank’.

Graphical Model (Notation)



- Nodes are random variables
- Edges denote possible dependence
- Observed variables are shaded
- Plates denote replicated structure

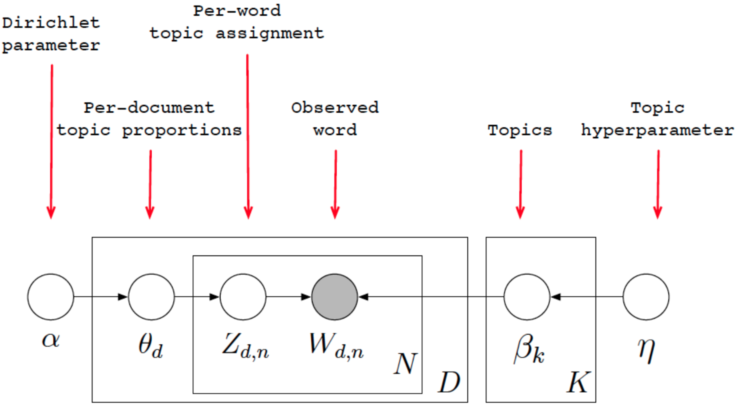
Graphical Model (Notation)



- Structure of the graph defines the pattern of conditional dependence between the ensemble of random variables
- E.g., this graph corresponds to

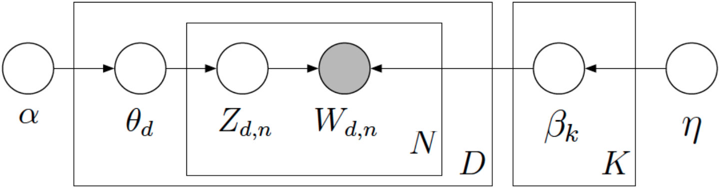
$$p(y, x_1, \ldots, x_N) = p(y) \prod_{n=1}^{N} p(x_n | y)$$

LDA: Graphical Model



Each piece of the structure is a random variable.

Latent Dirichlet Allocation: Generative Model



1. Draw each topic $\beta_i \sim \mathrm{Dir}(\eta)$, for $i \in \{1, \ldots, K\}$.
2. For each document:
   1. Draw topic proportions $\theta_d \sim \mathrm{Dir}(\alpha)$.
   2. For each word:
      1. Draw $Z_{d,n} \sim \mathrm{Mult}(\theta_d)$.
      2. Draw $W_{d,n} \sim \mathrm{Mult}(\beta_{Z_{d,n}})$.

What is Latent Dirichlet Allocation (LDA)?

- 'Latent' has the same sense in LDA as in Latent semantic indexing, i.e. capturing topics as latent variables
- The distribution that is used to draw the per-document topic distributions is called a *Dirichlet distribution*. This result is used to allocate the words of the documents to different topics.

What is Latent Dirichlet Allocation (LDA)?

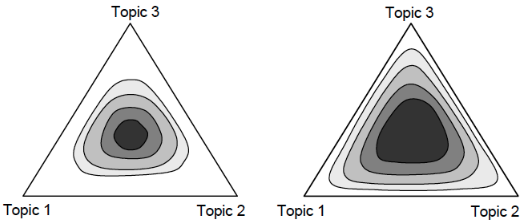
- 'Latent' has the same sense in LDA as in Latent semantic indexing, i.e. capturing topics as latent variables
- The distribution that is used to draw the per-document topic distributions is called a *Dirichlet distribution*. This result is used to allocate the words of the documents to different topics.

Dirichlet Distribution

The Dirichlet distribution is an exponential family distribution over the simplex, i.e. positive vectors that sum to one

$$p(\theta \mid \vec{\alpha}) = \frac{\Gamma\left(\sum_i \alpha_i\right)}{\prod_i \Gamma(\alpha_i)} \prod_i \theta_i^{\alpha_i - 1}$$

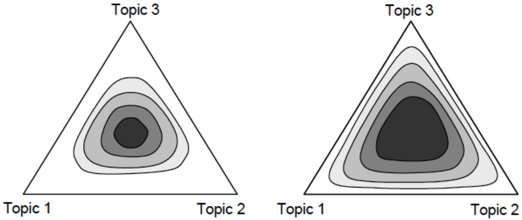
$$p(\theta \mid \vec{\alpha}) = \frac{\Gamma\left(\sum_i \alpha_i\right)}{\prod_i \Gamma(\alpha_i)} \prod_i \theta_i^{\alpha_i - 1}$$



α_is: **hyper-parameters of the model:**

α_j can be interpreted as a prior observation count for the number of times topic j is sampled in a document

$$p(\theta \mid \vec{\alpha}) = \frac{\Gamma\left(\sum_i \alpha_i\right)}{\prod_i \Gamma(\alpha_i)} \prod_i \theta_i^{\alpha_i - 1}$$

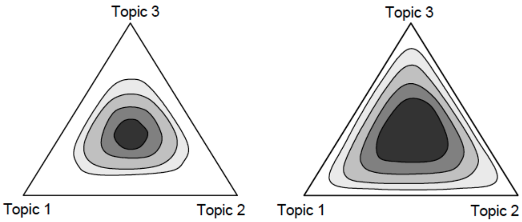


α_is: **hyper-parameters of the model:**

These priors can be interpreted as forces in the topic distributions with higher α moving the topics away from the corners of the simplex

Dirichlet Distribution

$$p(\theta \mid \vec{\alpha}) = \frac{\Gamma\left(\sum_i \alpha_i\right)}{\prod_i \Gamma(\alpha_i)} \prod_i \theta_i^{\alpha_i - 1}$$

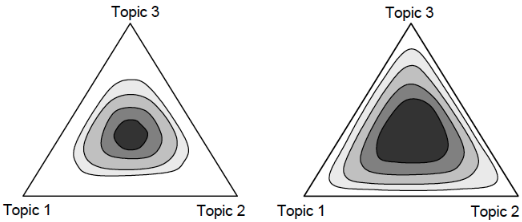


α_is: **hyper-parameters of the model:**

When $\alpha < 1$, there is a bias to pick topic distributions favoring just a few topics

Dirichlet Distribution

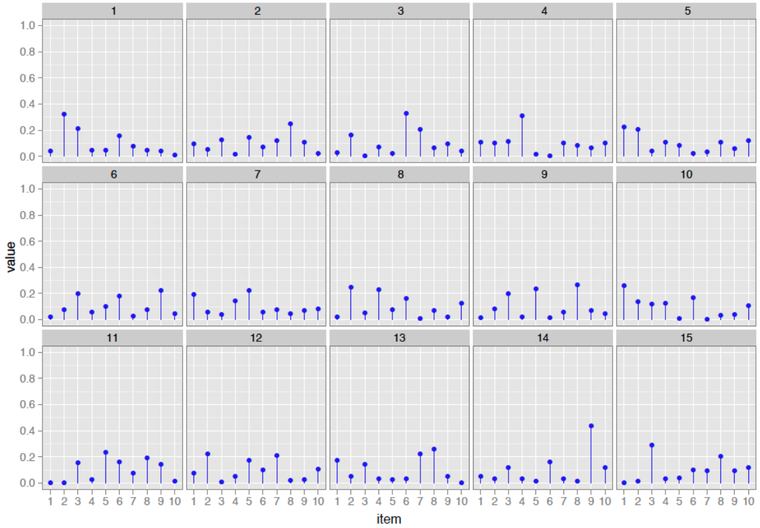
$$p(\theta \mid \vec{\alpha}) = \frac{\Gamma\left(\sum_i \alpha_i\right)}{\prod_i \Gamma(\alpha_i)} \prod_i \theta_i^{\alpha_i - 1}$$



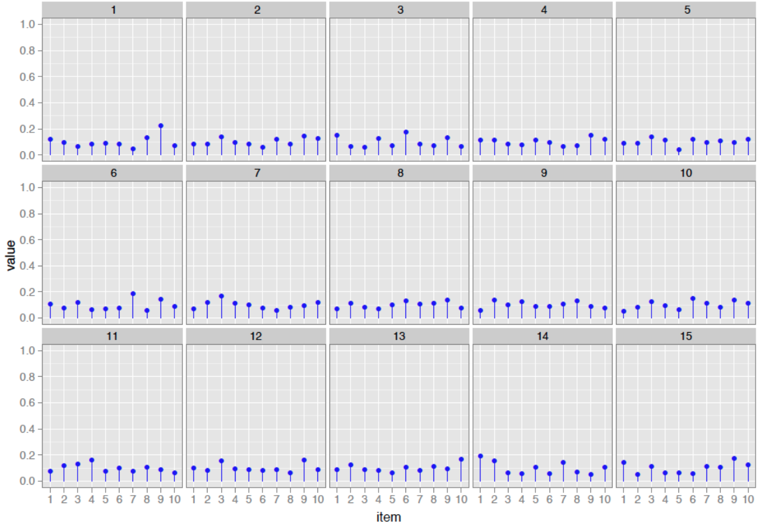
α_is: **hyper-parameters of the model:**

It is convenient to use a symmetric Dirichlet distribution with a single hyper-parameter $\alpha_1 = \alpha_2 \ldots = \alpha$

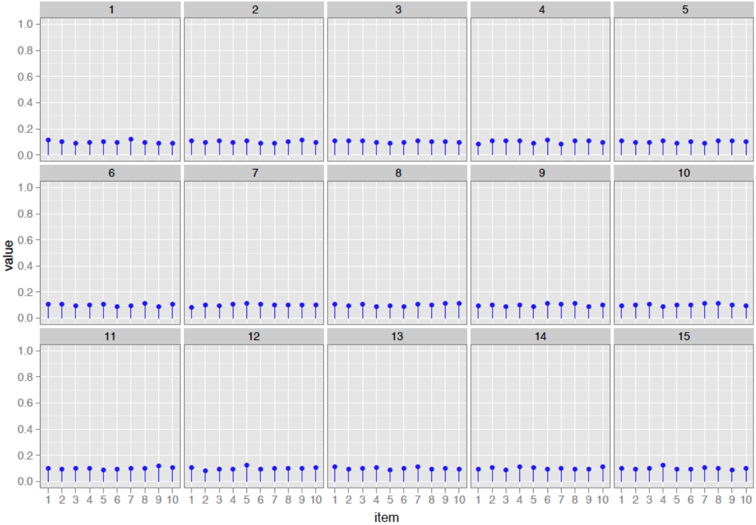
Effect of α: $\alpha = 1$



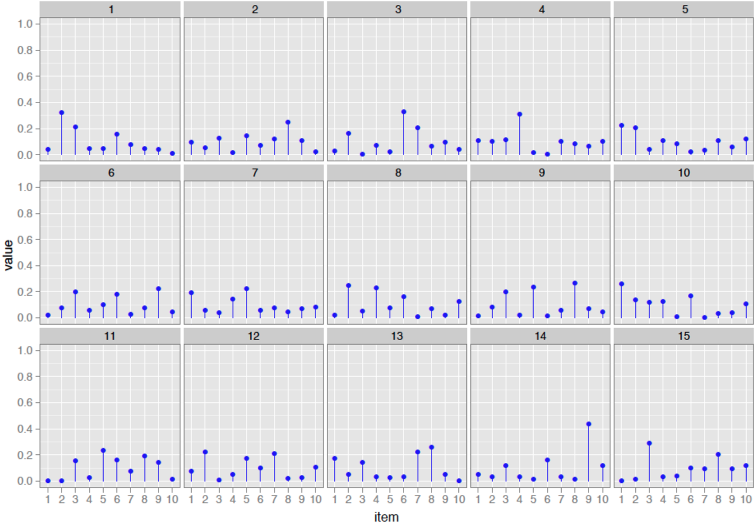
Effect of α: $\alpha = 10$



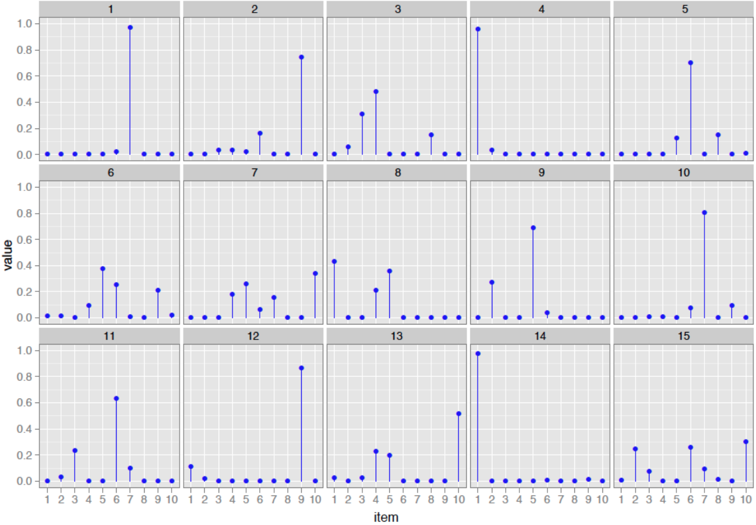
Effect of α: $\alpha = 100$



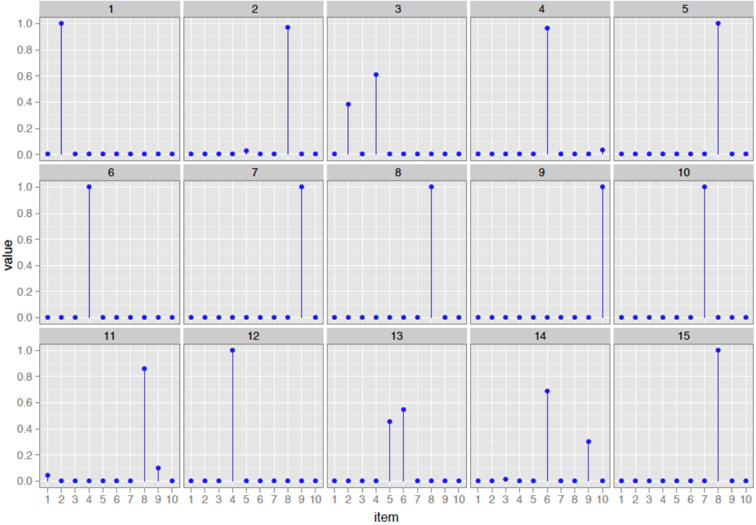
Effect of α: $\alpha = 1$



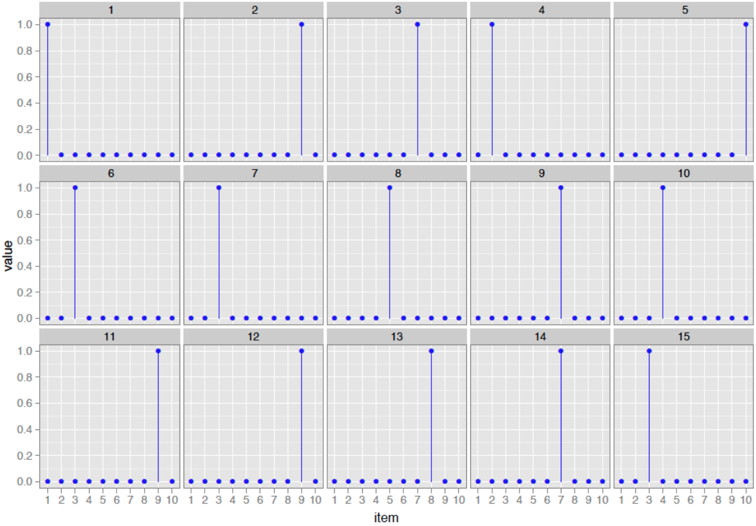
Effect of α: $\alpha = 0.1$



Effect of α: $\alpha = 0.01$



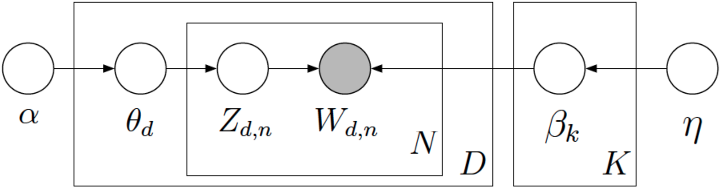
Effect of α: $\alpha = 0.001$



Online Implementations

LDA-C*	A C implementation of LDA
HDP*	A C implementation of the HDP ("infinite LDA")
Online LDA*	A python package for LDA on massive data
LDA in R*	Package in R for many topic models
LingPipe	Java toolkit for NLP and computational linguistics
Mallet	Java toolkit for statistical NLP
TMVE*	A python package to build browsers from topic models

Latent Dirichlet Allocation: Statistical Inference



- 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
- Use posterior expectations to perform the task at hand, e.g., information retrieval, document similarity, etc.

Approximating the posterior

Algorithms to approximate it fall in two categories:

Sampling-based Algorithms

Collect samples from the posterior to approximate it with an empirical distribution

Approximating the posterior

Algorithms to approximate it fall in two categories:

Sampling-based Algorithms

Collect samples from the posterior to approximate it with an empirical distribution

Variational Methods

- Deterministic alternative to sampling-based algorithms
- The inference problem is transformed to an optimization problem

Gibbs Sampling

- A form of Markov chain Monte Carlo (MCMC), which simulates a high-dimensional distribution by sampling on lower-dimensional subset of variables where each subset is conditioned on the value of all others
- Sampling is done sequentially and proceeds until the sampled values approximate the target distribution
- It directly estimates the posterior distribution over z , and uses this to provide estimates for β and θ

Gibbs Sampling

- Suppose we have a word token i for which we want to find the topic assignment probability : $p(z_i = j)$
- Represent the collection of documents by a set of word indices w_i and document indices d_i for this token i
- Gibbs sampling considers each word token in turn and estimates the probability of assigning the current word token to each topic, conditioned on the topic assignment to all other word tokens
- From this conditional distribution, a topic is sampled and stored as the new topic assignment for this word token
- This conditional is written as $P(z_i = j | z_{-i}, w_i, d_i, .)$

Gibbs Sampling

- Let us define two matrices C^{WT} and C^{DT} of dimensions $W \times T$ and $D \times T$ respectively.
- ${C_{wj}}^{WT}$ contains the number of times word w is assigned to topic j, not including the current instance
- ${C_{dj}}^{WT}$ contains the number of times topic j is assigned to some word token in document d, not including the current instance

Gibbs Sampling

- Let us define two matrices C^{WT} and C^{DT} of dimensions $W \times T$ and $D \times T$ respectively.
- ${C_{wj}}^{WT}$ contains the number of times word w is assigned to topic j, not including the current instance
- ${C_{dj}}^{WT}$ contains the number of times topic j is assigned to some word token in document d, not including the current instance

$$P(z_i = j | z_{-i}, w_i, d_i, .) \propto \frac{{C_{w_i j}}^{WT} + \eta}{\sum_{w=1}^{W} {C_{wj}}^{WT} + W\eta} \frac{{C_{d_i j}}^{DT} + \alpha}{\sum_{t=1}^{T} {C_{dj}}^{DT} + T\alpha}$$

- The left part is the probability of word w under topic j (How likely a word is for a topic) whereas
- the right part is the probability of topic j under the current topic distribution for document d (How dominant a topic is in a document)

Algorithm

- Start: Each word token is assigned to a random topic in $[1 \ldots T]$
- For each word token, a new topic is sampled as per $P(z_i = j|z_{-i}, w_i, d_i, .)$, adjusting the matrices C^{WT} and C^{DT}
- A single pass through all word tokens in the document is one *Gibbs sample*
- After the burnin period, these samples are saved at regularly spaced intervals, to prevent correlations between samples

Estimating θ and β

$$\beta_i^{(j)} = \frac{C_{ij}{}^{WT} + \eta}{\sum_{k=1}^{W} C_{kj}{}^{WT} + W\eta}$$

$$\theta_j^{(d)} = \frac{C_{dj}{}^{DT} + \alpha}{\sum_{k=1}^{T} C_{dk}{}^{DT} + T\alpha}$$

These values correspond to predictive distributions of

- sampling a new token of word i from topic j, and
- sampling a new token in document d from topic j

An Example

The algorithm can be illustrated by generating artificial data from a known topic model and applying the algorithm to check whether it is able to infer the original generative structure.

Example

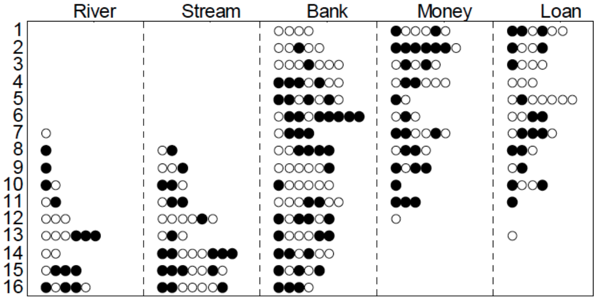
- Let topic 1 give equal probability to MONEY, LOAN, BANK and topic 2 give equal probability to words RIVER, STREAM, and BANK

$$\beta_{MONEY}{}^{(1)} = \beta_{LOAN}{}^{(1)} = \beta_{BANK}{}^{(1)} = 1/3$$

$$\beta_{RIVER}{}^{(2)} = \beta_{STREAM}{}^{(2)} = \beta_{BANK}{}^{(2)} = 1/3$$

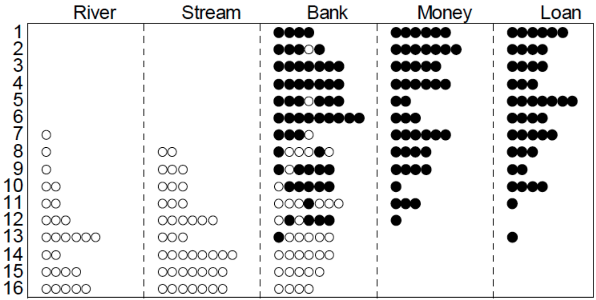
- We generate 16 documents by arbitrarily mixing two topics.

Initial Structure



Colors reflect initial random assignment, black = topic 1, while = topic 2

After 64 iterations of Gibbs Sampling



$$\beta_{MONEY}{}^{(1)} = 0.32, \beta_{LOAN}{}^{(1)} = 0.29, \beta_{BANK}{}^{(1)} = 0.39$$

$$\beta_{RIVER}{}^{(2)} = 0.25, \beta_{STREAM}{}^{(2)} = 0.4, \beta_{BANK}{}^{(2)} = 0.35$$

Computing Similarities

Document Similarity

Similarity between documents d_i and d_2 can be measured by the similarity between their topic distributions $\theta^{(d_1)}$ and $\theta^{(d_2)}$

KL divergence : $D(p,q) = \sum_{j=1}^{T} p_j log_2 \frac{p_j}{q_j}$

Symmetrized KL divergence: $\frac{1}{2}[D(p,q)+D(q,p)]$ seems to work well

Similarity with respect to query q

Maximize the conditional probability of query given the document:

$$p(q|d_i) = \prod_{w_k \in q} p(w_k|d_i)$$

$$= \prod_{w_k \in q} \sum_{j=1}^{T} P(w_k|z=j)P(z=j|d_i)$$

Computing Similarities

Similarity between two words

Having observed a single word in a new context, what are the other words that might appear in the same context, based on the topic interpretation for the observed word?

$$p(w_2|w_1) = \sum_{j=1}^{T} p(w_2|z=j)p(z=j|w_i)$$

Example

Observed and predicted responses for the word 'PLAY'

HUMANS		TOPICS	
FUN	.141	BALL	.036
BALL	.134	GAME	.024
GAME	.074	CHILDREN	.016
WORK	.067	TEAM	.011
GROUND	.060	WANT	.010
MATE	.027	MUSIC	.010
CHILD	.020	SHOW	.009
ENJOY	.020	HIT	.009
WIN	.020	CHILD	.008
ACTOR	.013	BASEBALL	.008
FIGHT	.013	GAMES	.007
HORSE	.013	FUN	.007
KID	.013	STAGE	.007
MUSIC	.013	FIELD	.006

Modeling Science

Data

The OCR'ed collection of *Science* from 1990-2000

- 17K documents
- 11M words
- 20K unique terms (stop words and rare words removed)

Modeling Science

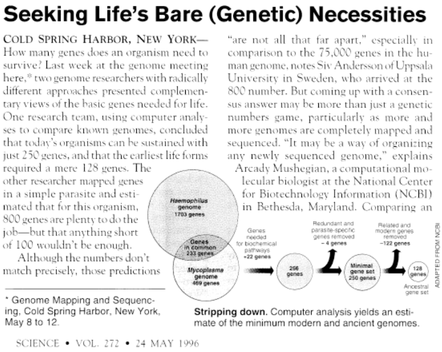
Data

The OCR'ed collection of *Science* from 1990-2000

- 17K documents
- 11M words
- 20K unique terms (stop words and rare words removed)

Model

100-topic model using variational inference



Seeking Life's Bare (Genetic) Necessities

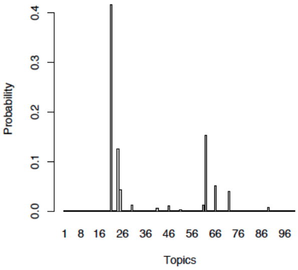
COLD SPRING HARBOR, NEW YORK— How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known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

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

* 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



Example Topics

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

Modeling Richer Assumptions in Topic Models

- Correlated topic models
- Dynamic topic models
- Measuring scholarly impact

Correlated Topic Models

- The Dirichlet is an exponential family distribution on the simplex, positive vectors that sum to one
- However, the near independence of components makes it a poor choice for modeling topic proportions
- An article about *fossil fuels* is more likely to also be about *geology* than about *genetics*

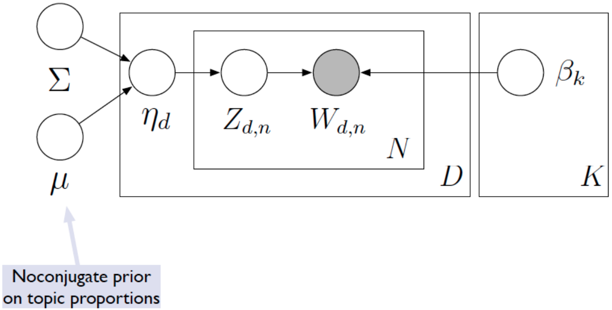
Using logistic normal distribution

A multivariate normal distribution of a k-dimensional vector $x = [X_1, X_2, \ldots, X_k]$ can be written as

$$x \sim N_k(\mu, \Sigma)$$

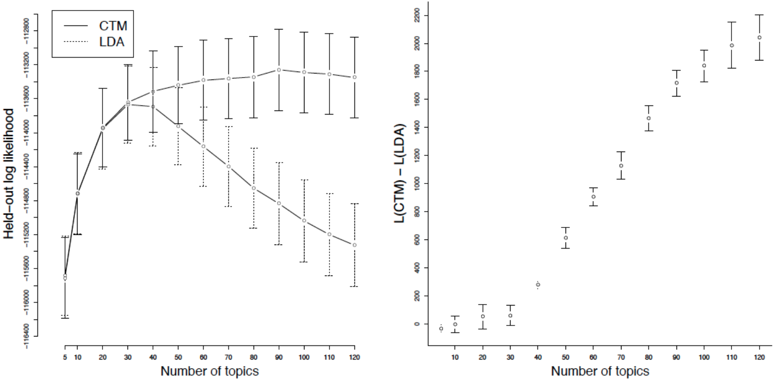
with k-dimensional mean vector μ and $k \times k$ covariance matrix Σ

Correlated Topic Model (CTM)



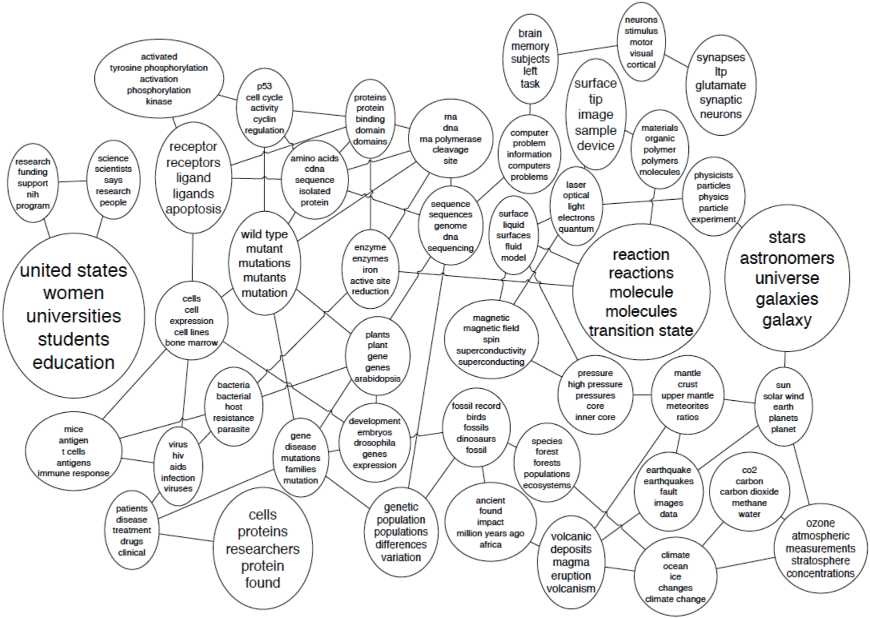
- Draw topic proportions from a logistic normal, where topic occurrences can exhibit correlation.
- Use for:
  - Providing a “map” of topics and how they are related
  - Better prediction via correlated topics

CTM supports more topics and provides a better fit than LDA



Held-out log probability on *Science*

Correlated Topics

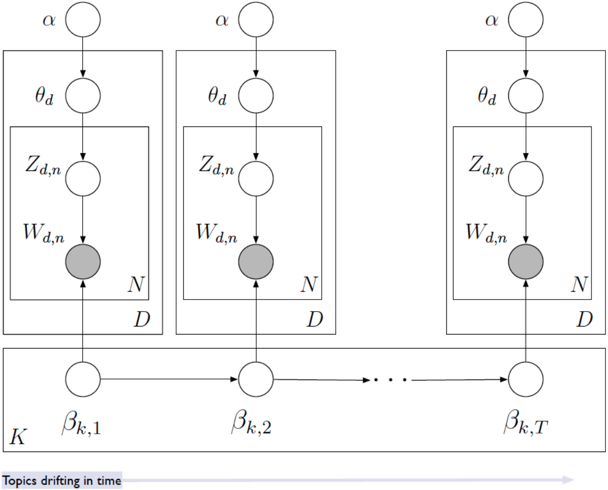


Dynamic Topic Models

LDA assumption

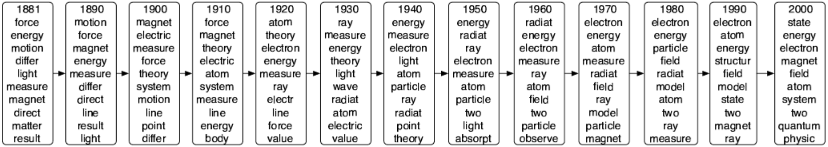
- LDA assumes that the order of documents does not matter
- Not appropriate for corpora that spans hundreds of years
- We might want to track how language changes over time

Dynamic Topic Models

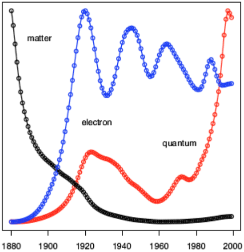


$$\beta_{k,t} | \beta_{k,t-1} \sim N(\beta_{k,t-1}, \sigma^2 I)$$

Dynamic Topic Models

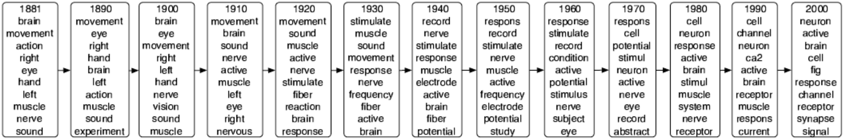


"Atomic Physics"

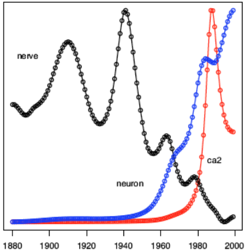


1881 On Matter as a form of Energy
1892 Non-Euclidean Geometry
1900 On Kathode Rays and Some Related Phenomena
1917 ``Keep Your Eye on the Ball"
1920 The Arrangement of Atoms in Some Common Metals
1933 Studies in Nuclear Physics
1943 Aristotle, Newton, Einstein. II
1950 Instrumentation for Radioactivity
1965 Lasers
1975 Particle Physics: Evidence for Magnetic Monopole Obtained
1985 Fermilab Tests its Antiproton Factory
1999 Quantum Computing with Electrons Floating on Liquid Helium

Dynamic Topic Models



"Neuroscience"



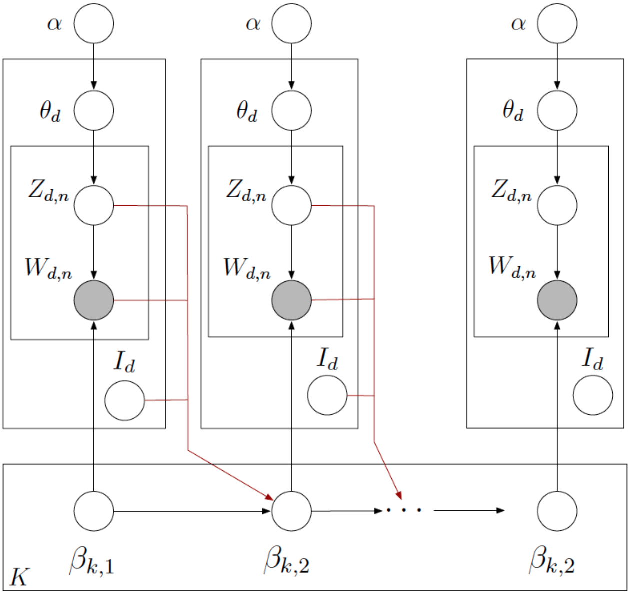
1887 Mental Science
1900 Hemianopsia in Migraine
1912 A Defence of the ``New Phrenology"
1921 The Synchronal Flashing of Fireflies
1932 Myoesthesis and Imageless Thought
1943 Acetylcholine and the Physiology of the Nervous System
1952 Brain Waves and Unit Discharge in Cerebral Cortex
1963 Errorless Discrimination Learning in the Pigeon
1974 Temporal Summation of Light by a Vertebrate Visual Receptor
1983 Hysteresis in the Force-Calcium Relation in Muscle
1993 GABA-Activated Chloride Channels in Secretory Nerve Endings

Measuring Scholarly Impact

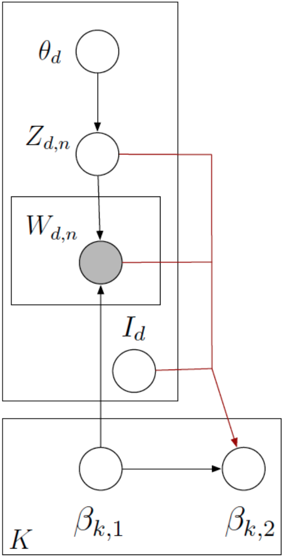
How to model influence?

- *Idea from Dynamic Topic Models*, influential articles reflect future changes in language use
- The *influence* of an article is a latent variable
- Influential articles affect the drift of the topics that they discuss
- The posterior gives a retrospective estimate of influential article

Measuring Scholarly Impact



Measuring Scholarly Impact



- Each document has an influence score I_d.
- Each topic drifts in a way that is biased towards the documents with high influence.
- The posterior of $I_{1:D}$ can be examined to retrospectively find articles that best explain future changes in language.

Supervised settings of LDA

Use data points paired with response variables

- User reviews paired with a number of stars
- Web pages paired with a number of likes
- Documents paired with links to other documents
- Images paired with a category

Supervised settings of LDA

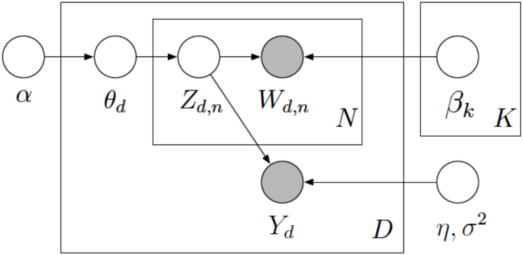
Use data points paired with response variables

- User reviews paired with a number of stars
- Web pages paired with a number of likes
- Documents paired with links to other documents
- Images paired with a category

Supervised topic modes

are topic models of documents and responses, fit to find topics predictive of the response

Supervised LDA



1. Draw topic proportions $\theta \mid \alpha \sim \mathrm{Dir}(\alpha)$.
2. For each word
   - Draw topic assignment $z_n \mid \theta \sim \mathrm{Mult}(\theta)$.
   - Draw word $w_n \mid z_n, \beta_{1:K} \sim \mathrm{Mult}(\beta_{z_n})$.
3. Draw response variable $y \mid z_{1:N}, \eta, \sigma^2 \sim \mathrm{N}(\eta^\top \bar{z}, \sigma^2)$, where

$$\bar{z} = (1/N) \sum_{n=1}^{N} z_n.$$

Supervised LDA: why a different model is required?

Think of an alternative approach using original settings of LDA

Supervised LDA: why a different model is required?

Think of an alternative approach using original settings of LDA

Formulate a model in which the response variable y is regressed on topic proportions θ

Supervised LDA: why a different model is required?

Think of an alternative approach using original settings of LDA

Formulate a model in which the response variable y is regressed on topic proportions θ

Why then a different model?

Supervised LDA: why a different model is required?

Think of an alternative approach using original settings of LDA

Formulate a model in which the response variable y is regressed on topic proportions θ

Why then a different model?

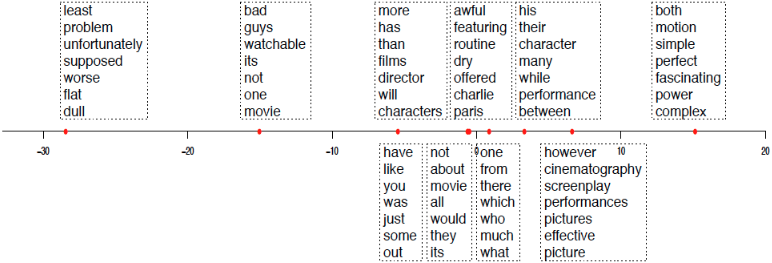
- The response variable can be treated as an important observation to infer the topic probabilities in a supervised manner

Prediction

- Fit sLDA parameters to documents and responses. This gives:
  - topics $\beta_{1:K}$
  - coefficients $\eta_{1:K}$
- We have a new document $w_{1:N}$ with unknown response value.
- We predict y using the SLDA expected value:

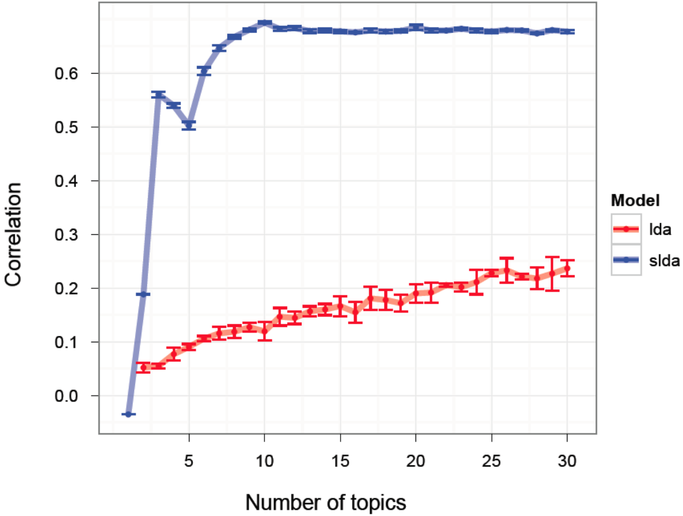
$$\mathrm{E}\left[Y \mid w_{1:N}, \alpha, \beta_{1:K}, \eta, \sigma^2\right] = \eta^{\top}\mathrm{E}\left[\bar{Z} \mid w_{1:N}\right]$$

Example: Movie Reviews

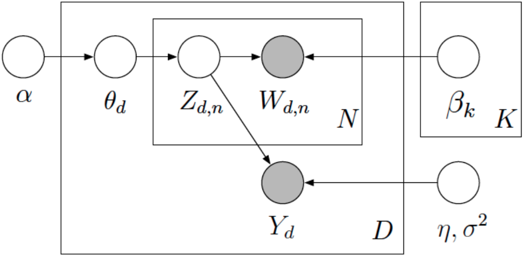


- 10-topic sLDA model on movie reviews (Pang and Lee, 2005).
- Response: number of stars associated with each review
- Each component of coefficient vector η is associated with a topic.

Held out correlation

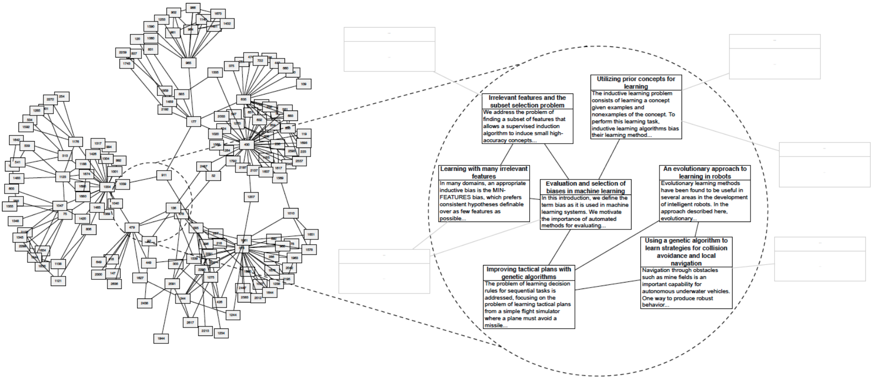


Supervised Topic Models



- SLDA enables model-based regression where the predictor "variable" is a text document.
- It can easily be used wherever LDA is used in an unsupervised fashion (e.g., images, genes, music).
- SLDA is a supervised dimension-reduction technique, whereas LDA performs unsupervised dimension reduction.

Relational Topic Models



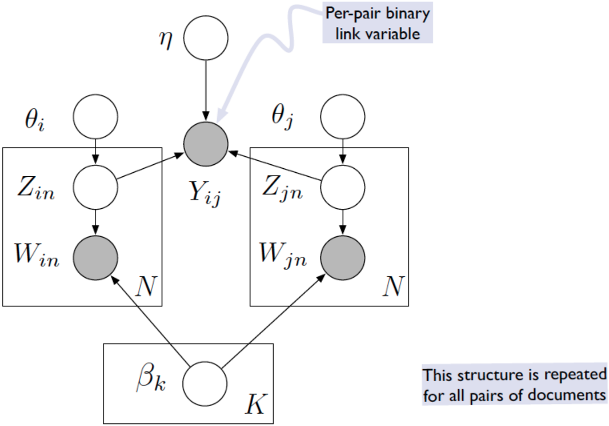
Connected Observations

- Citation networks of documents
- Hyperlinked networks of web-pages
- Friend-connected social network profiles

Relational Topic Models

- LDA needs to be adapted to a model of content and connection
- RTMs find hidden structure in both types of data

Relational Topic Models



Works in a supervised framework, allowing predictions about new and unlinked data

Link Prediction Task using RTM

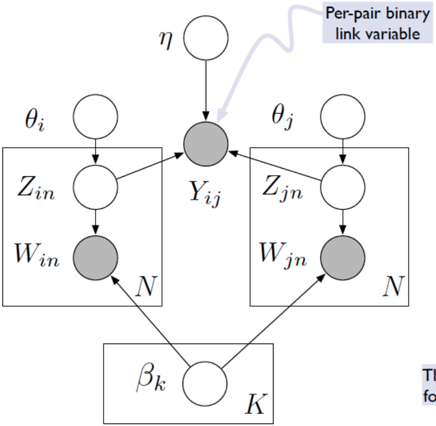
Given a new document, which documents is it likely to link to?

Markov chain Monte Carlo convergence diagnostics: A comparative review
Minorization conditions and convergence rates for Markov chain Monte Carlo
Rates of convergence of the Hastings and Metropolis algorithms
Possible biases induced by MCMC convergence diagnostics
Bounding convergence time of the Gibbs sampler in Bayesian image restoration
Self regenerative Markov chain Monte Carlo
Auxiliary variable methods for Markov chain Monte Carlo with applications
Rate of Convergence of the Gibbs Sampler by Gaussian Approximation
Diagnosing convergence of Markov chain Monte Carlo algorithms

RTM allows for such predictions

- links given the new words of a document
- words given the links of a new document

Relational Topic Models



This structure is repeated for all pairs of documents

Formulation ensures that the same latent topic assignments used to generate the content of the documents also generates their link structure.

RTM: Generative Model

1. For each document d:
   (a) Draw topic proportions $\theta_d|\alpha \sim \text{Dir}(\alpha)$.
   (b) For each word $w_{d,n}$:
      i. Draw assignment $z_{d,n}|\theta_d \sim \text{Mult}(\theta_d)$.
      ii. Draw word $w_{d,n}|z_{d,n}, \beta_{1:K} \sim \text{Mult}(\beta_{z_{d,n}})$.
2. For each pair of documents d, d':
   (a) Draw binary link indicator

$$y|\boldsymbol{z}_d, \boldsymbol{z}_{d'} \sim \psi(\cdot|\boldsymbol{z}_d, \boldsymbol{z}_{d'}).$$

Link Probability Function (ψ)

Dependent on the topic assignments that generated their words, z_d and $z_{d'}$.

$$\psi_e(y = 1) = exp(\eta^T(\overline{z_d} \circ \overline{z_{d'}}) + \nu)$$

- $z_d = \frac{1}{N_d} \sum_n z_{d,n}$
- $\circ$ notation denotes the Hadamard (element-wise) product
- Exponential function is being used, they also tried using sigmoid (ψ_σ)
- Link function models each per-pair binary variable as a logistic regression parameterized by $\eta_{1\times K}$ and intercept ν (in case of sigmoid)
- Covariates are constructed by the Hadamard product of $\overline{z_d}$ and $\overline{z_{d'}}$, capturing similarity between the hidden topic representation of the two documents

Inference: How many links to model

- One can fix $y_{d_1,d_2} = 1$ whenever a link is observed between d_1 and d_2 and set $y_{d_1,d_2} = 0$ otherwise

Inference: How many links to model

- One can fix $y_{d_1,d_2} = 1$ whenever a link is observed between d_1 and d_2 and set $y_{d_1,d_2} = 0$ otherwise
- Problem with that approach is that the absence of a link cannot be construed as evidence for $y_{d_1,d_2} = 0$
- So, in these cases, these links are treated as unobserved variables
- Also provides a significant computational advantage

In large social networks like Facebook, the absence of a link between two people doesn't necessarily mean that they are not friends.

Datasets: Summary Statistics

Data set	# of documents	# of words	Number of links	Lexicon size
Cora	2708	49216	5278	1433
WebKB	877	79365	1388	1703
PNAS	2218	11,9162	1577	2239
LocalNews	51	93765	107	1242

Preprocessing

Stop-words were removed and directed links were converted to undirected links, documents with no links were removed.

Datasets: Summary Statistics

What each dataset is about?

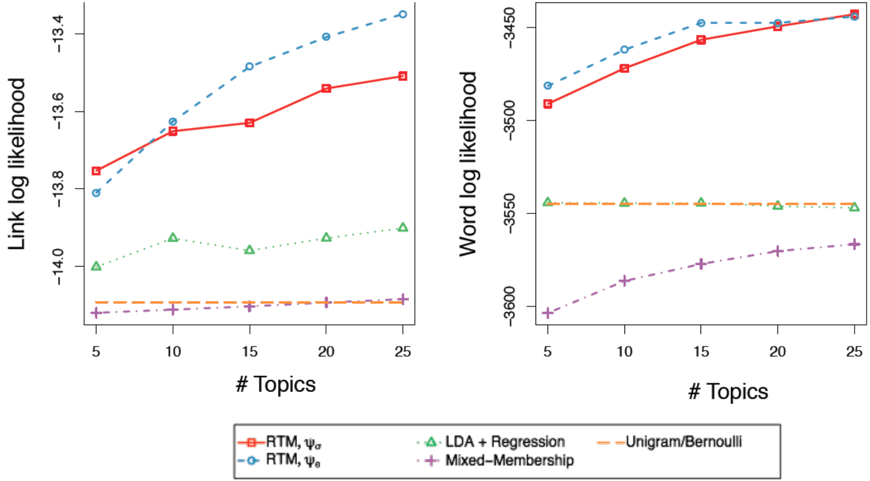
Cora Dataset contains abstracts from the Cora computer science research paper search engine, with links between documents that cite each other

WebKB Dataset contains web pages from the computer science departments of different universities, with links determined from the hyperlinks on each page

PNAS Dataset contains recent abstracts from PNAS. Links between the documents are intra-PNAS citations

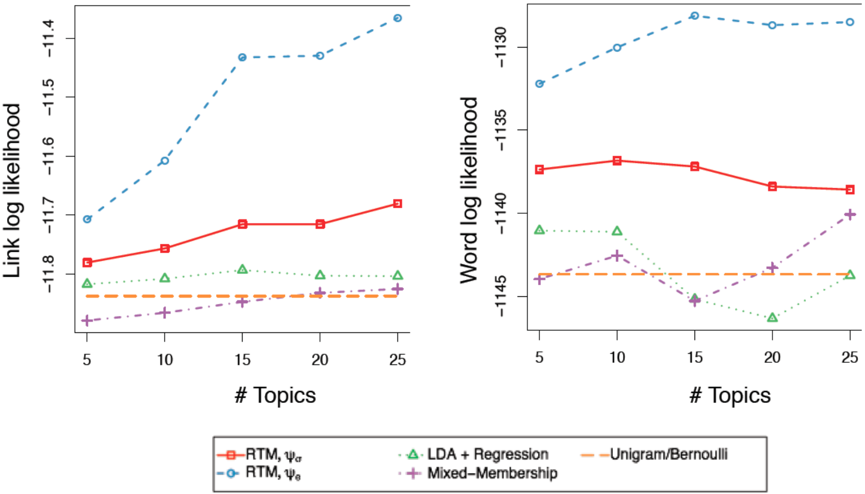
LocalNews Dataset is a corpus of local news culled from various media markets throughout the US. One document for each state, consisting of headlinesa dn summaries from local news. Links determined by geographical adjacency.

Predictve Performance



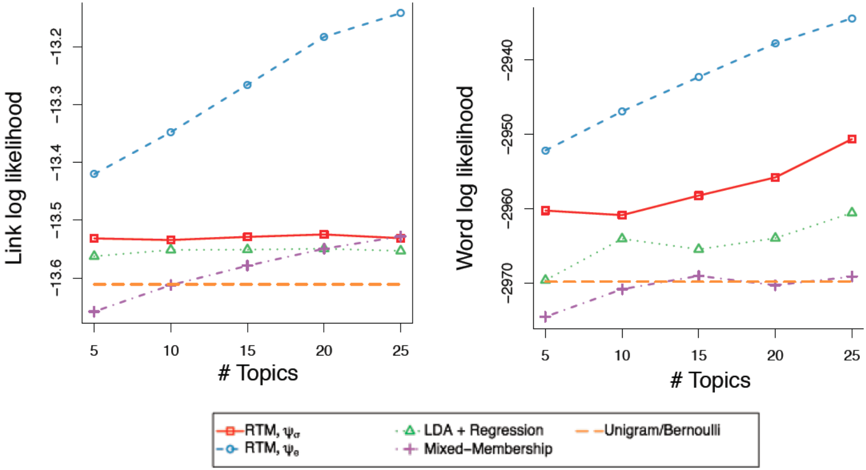
Cora corpus (McCallum et al., 2000)

Predictve Performance



WebKB corpus (Craven et al., 1998)

Predictve Performance



PNAS corpus (courtesy of JSTOR)

Predicting links from documents

Markov chain Monte Carlo convergence diagnostics: A comparative review	
Minorization conditions and convergence rates for Markov chain Monte Carlo	RTM (ψ_e)
Rates of convergence of the Hastings and Metropolis algorithms	
Possible biases induced by MCMC convergence diagnostics	
Bounding convergence time of the Gibbs sampler in Bayesian image restoration	
Self regenerative Markov chain Monte Carlo	
Auxiliary variable methods for Markov chain Monte Carlo with applications	
Rate of Convergence of the Gibbs Sampler by Gaussian Approximation	
Diagnosing convergence of Markov chain Monte Carlo algorithms	
Exact Bound for the Convergence of Metropolis Chains	LDA + Regression
Self regenerative Markov chain Monte Carlo	
Minorization conditions and convergence rates for Markov chain Monte Carlo	
Gibbs-markov models	
Auxiliary variable methods for Markov chain Monte Carlo with applications	
Markov Chain Monte Carlo Model Determination for Hierarchical and Graphical Models	
Mediating instrumental variables	
A qualitative framework for probabilistic inference	
Adaptation for Self Regenerative MCMC	

Given a new document, which documents is it likely to link to?

Predicting links from documents

Competitive environments evolve better solutions for complex tasks	
Coevolving High Level Representations A Survey of Evolutionary Strategies **Genetic Algorithms in Search, Optimization and Machine Learning** **Strongly typed genetic programming in evolving cooperation strategies** Solving combinatorial problems using evolutionary algorithms A promising genetic algorithm approach to job-shop scheduling... Evolutionary Module Acquisition An Empirical Investigation of Multi-Parent Recombination Operators...	RTM (ψ_e)
A New Algorithm for DNA Sequence Assembly Identification of protein coding regions in genomic DNA Solving combinatorial problems using evolutionary algorithms A promising genetic algorithm approach to job-shop scheduling... A genetic algorithm for passive management The Performance of a Genetic Algorithm on a Chaotic Objective Function Adaptive global optimization with local search Mutation rates as adaptations	**LDA + Regression**

Given a new document, which documents is it likely to link to?

Example Problem

Suppose you are using Gibbs sampling to estimate the distributions, θ and β for topic models. The underlying corpus has 5 documents and 5 words, $\{$*River, Stream, Bank, Money, Loan*$\}$ and the number of topics is 2. At certain point, the structure of the documents looks like the following Table. For instance, the first row indicates that the document 1 contains 4 instances of word 'Bank', 6 instances of word 'Money' and 6 instances of word 'Loan'. Black and white circles denote whether the word is currently assigned to topics t_1 and t_2 respectively.

Use this structure to estimate $\beta_{MONEY}{}^{(2)}$ and $\beta_{BANK}{}^{(1)}$ at this point. You can take the values of η and α to be 0.1 each.

Doc. Id	River	Stream	Bank	Money	Loan
1			●●●●	●●●●●●	●●●●●●
2			●●●○●	●●●●●●●	●●●●
3	○	○○○	●○○○●○	●●●●	●●●
4	○○○○○○	○○○	●○○○○○		
5	○○	○○○○○○	○○○○○○		