Scalable Topic Models

David M. Blei

Department of Computer Science
Princeton University

January 25, 2011
As more information becomes available, it becomes more difficult to find and discover what we need.

We need new tools to help us organize, search, and understand these vast amounts of information.
Topic modeling provides methods for automatically organizing, understanding, searching, and summarizing large electronic archives.

1. Discover the hidden themes that pervade the collection.
2. Annotate the documents according to those themes.
3. Use annotations to organize, summarize, and search the texts.
Discover topics from a corpus

- human
- genome
- dna
- genetic
- genes
- sequence
- gene
- molecular
- sequencing
- map
- information
- genetics
- mapping
- project
- sequences

- evolution
- evolutionary
- species
- organisms
- life
- origin
- biology
- groups
- phylogenetic
- living
- diversity
- group
- new
- two
- common

- disease
- host
- bacteria
- diseases
- resistance
- bacterial
- new
- strains
- control
- infectious
- malaria
- parasite
- parasites
- united
- tuberculosis

- computer
- models
- information
- data
- computers
- system
- network
- systems
- model
- parallel
- methods
- networks
- software
- new
- simulations
Model the evolution of topics over time

"Theoretical Physics"

"Neuroscience"
Model connections between topics
Browser a large archive in an organized way

Automatic Analysis, Theme Generation, and Summarization of Machine-Readable Texts

Gerard Salton, James Allan, Chris Buckley

Vast amounts of text material are now available in machine-readable form. The availability of machine-readable texts, and the fact that they are amenable to automatic processing, is a boon for those who need to access large amounts of text in a systematic way. Because the available databases are large and cover many different subject areas, automatic aids must be provided to users interested in accessing the data. It has been suggested that links be placed between related pieces of text, connecting, for example, particular text paragraphs to other paragraphs covering related subject matter. Such a linked text structure, often called hypertext, makes it possible for the reader to start with particular text passages and use the linked structure to find related text elements. Unfortunately, until now, viable methods for automating the building of hypertext structures and for using such structures in a sophisticated way have not been available. Here we give methods for constructing text relation maps and for using such maps to access and use text databases. In particular, we outline procedures for determining text themes, traversing texts selectively, and extracting summary statements that reflect text content.

Text Analysis and Retrieval: The Smart System

The Smart system is a sophisticated text retrieval tool, developed over the past 30 years, that is based on the vector space model of retrieval. It is based on the assumption that a document is a weighted linear combination of words, and that the similarity between two documents is determined by the cosine of the angle between their vectors. The Smart system uses a term-by-document matrix to represent the frequency of each term in each document, and weights the terms based on their importance. It then uses linear algebra to compute the similarity between documents, and uses this similarity to rank the documents in response to a query.
Collect a large archive of electronic texts.

Run a topic modeling algorithm on them.

Use the results to explore, analyze and browse the archive.
Scalable topic modeling

- Existing topic modeling algorithms process document collections in **batch**—iteratively examining each document.

- Many applications of topic modeling could benefit from processing documents in a **stream**:
  - Linking topic models to web APIs and databases
  - Handling millions and billions of documents
  - Refining topic models on the fly, e.g., for user interfaces
This talk

1. An introduction to topic modeling
2. A variety of topic models
3. Scalable topic models
Introduction to topic modeling
**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

---


---

**Simple intuition:** Documents exhibit multiple topics.
Generative model for latent Dirichlet allocation (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
In reality, we only observe the documents

The other structure are hidden variables
Our goal is to **infer** the hidden variables

I.e., compute their distribution conditioned on the documents

\[ p(\text{topics}, \text{proportions}, \text{assignments} | \text{documents}) \]
LDA as a graphical model

- Vertices denote random variables.
- Edges denote dependence between random variables.
- Shading denotes observed variables.
- Plates denote replicated variables.
LDA as a graphical model

This represents a factorization of the joint distribution

$$\prod_{i=1}^{K} p(\beta_i | \eta) \prod_{d=1}^{D} p(\theta_d | \alpha) \left( \prod_{n=1}^{N} p(z_{d,n} | \theta_d) p(w_{d,n} | \beta_1:K, z_{d,n}) \right)$$
Latent Dirichlet allocation

• From a collection of documents, infer the posterior of
  • per-word topic assignment $z_{d,n}$
  • per-document topic proportions $\theta_d$
  • per-corpus topic distributions $\beta_k$

• Use this posterior to perform the task at hand, e.g., information retrieval, document similarity, and others.
**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 LDA model using variational inference.
Example 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


Striping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.
Chaotic Beetles

Charles Godfray and Michael Hassell

Ecologists have known since the pioneering work of May in the mid-1970s (1) that the population dynamics of animals and plants can be exceedingly complex. This complexity arises from two sources: The tangled web of interactions that constitute any natural community provide a myriad of different pathways for species to interact, both directly and indirectly. And even in isolated populations the nonlinear feedback processes present in all natural populations can result in complex dynamic behavior. Natural populations can show persistent oscillatory dynamics and chaos, the latter characterized by extreme sensitivity to initial conditions. If such chaotic dynamics were common in nature, then this would have important ramifications for the management and conservation of natural resources. On page 389 of this issue, Costantino et al. (2) provide the most convincing evidence to date of complex dynamics and chaos in a biological population—of the flour beetle, Tribolium castaneum (see figure).

It has proven extremely difficult to demonstrate complex dynamics in populations in the field. By its very nature, a chaotically fluctuating population will superficially resemble a stable or cyclic population buffered by the normal random perturbations experienced by all species. Given a long enough time series, diagnostic tools from nonlinear mathematics can be used to identify the tell-tale signatures of chaos. In phase space, chaotic trajectories come to lie on “strange attractors,” curious geometric objects with fractal structure and hence noninteger dimension. As they move over the surface of the attractor, sets of adjacent trajectories are pulled apart, then stretched and folded, so that it becomes impossible to predict exact population densities into the future. The strength of the mixing that gives rise to the extreme sensitivity to initial conditions can be measured mathematically estimating the Liapunov exponent, which is positive for chaotic dynamics and nonpositive otherwise. There have been many attempts to estimate attractor dimension and Liapunov exponents from time series data, and some candidate chaotic population have been identified (some insects, rodents, and most convincingly, human childhood diseases), but the statistical difficulties preclude any broad generalization (3).

An alternative approach is to parameterize population models with data from natural populations and then compare their predictions with the dynamics in the field. This technique has been gaining popularity in recent years, helped by statistical advances in parameter estimation. Good ex-
<table>
<thead>
<tr>
<th>problem</th>
<th>model</th>
<th>selection</th>
<th>species</th>
</tr>
</thead>
<tbody>
<tr>
<td>problems</td>
<td>rate</td>
<td>male</td>
<td>forest</td>
</tr>
<tr>
<td>mathematical</td>
<td>constant</td>
<td>males</td>
<td>ecology</td>
</tr>
<tr>
<td>number</td>
<td>distribution</td>
<td>females</td>
<td>fish</td>
</tr>
<tr>
<td>new</td>
<td>time</td>
<td>sex</td>
<td>ecological</td>
</tr>
<tr>
<td>mathematics</td>
<td>number</td>
<td>species</td>
<td>conservation</td>
</tr>
<tr>
<td>university</td>
<td>size</td>
<td>female</td>
<td>diversity</td>
</tr>
<tr>
<td>two</td>
<td>values</td>
<td>evolution</td>
<td>population</td>
</tr>
<tr>
<td>first</td>
<td>value</td>
<td>populations</td>
<td>natural</td>
</tr>
<tr>
<td>numbers</td>
<td>average</td>
<td>population</td>
<td>ecosystems</td>
</tr>
<tr>
<td>work</td>
<td>rates</td>
<td>sexual</td>
<td>populations</td>
</tr>
<tr>
<td>time</td>
<td>data</td>
<td>behavior</td>
<td>endangered</td>
</tr>
<tr>
<td>mathematicians</td>
<td>density</td>
<td>evolutionary</td>
<td>tropical</td>
</tr>
<tr>
<td>chaos</td>
<td>measured</td>
<td>genetic</td>
<td>forests</td>
</tr>
<tr>
<td>chaotic</td>
<td>models</td>
<td>reproductive</td>
<td>ecosystem</td>
</tr>
</tbody>
</table>
Automatic Analysis, Theme Generation, and Summarization of Machine-Readable Texts

Gerard Salton, James Allan, Chris Buckley

Many kinds of texts are currently available in machine-readable form and are amenable to automatic processing. Because the available databases are large and cover many different subject areas, automatic aids must be provided to users interested in accessing the data. It has been suggested that links be placed between related pieces of text, connecting, for example, particular text paragraphs or other paragraphs covering related subject matter. Such a linked text structure, often called hypertext, makes it possible for the reader to start with particular text passages and use the linked structure to find related text elements. Unfortunately, until now, viable methods for automatically building large hypertext structures and for using such structures in a sophisticated way have not been available. Here we give methods for constructing text relation maps and for using text relations to access and use text databases. In particular, we outline a procedure for determining text themes, traversing texts selectively, and extracting summary statements that reflect text content.

Text Analysis and Retrieval: The Smart System

The Smart system is a sophisticated text retrieval tool, developed over the past 20 years, that is based on the vector space model of retrieval, which involves as well as information sets, or is typically a word, associated with a topic. In principle, chosen from a context of a document, but by constructing such a context, an information retrieval system can derive the terms under consideration assigned to a text content.

Because the text for content representation introduces a term-word matrix, high weights to lower weights to, a powerful tool in the field is the well-known term frequency (or weight) frequency (or weight) of a term in a document with a low frequency (or weight) of a term in a document. Such terms distinguish the documents.

When all texts are represented by weighted D = 4da, dw, weight assigned to term frequency in a document to term frequency in a document. Thus, a model of retrieval based on the vector space model of retrieval is used to perform information retrieval.

Fig. 1. The match used for ranking the different models of matching the "T:A." field. The four columns of three lines indicate: (1) the pattern, (2) the number of words, (3) the number of words, and (4) the number of words. The results are shown in the table. For example, the model with the highest score is the model with the best score.

On reading Professor Minot's explanation of his method of storing pamphlets, given in the issues of December 2001 and 1994, I feel inclined to add a word in commendation of the method. I began using these boxes six or seven years ago and now have 160 upon my shelves. About one-half are devoted to Experiments Station bulletins, the boxes being labeled by States and arranged alphabetically. The other half is used for miscellaneous pamphlets on subjects pertaining to my line of work. This box has proven itself satisfactory in every way, and as a simple time-saving device to pass the time. My system of pamphlet arrangement differs in some respects from that adopted by Professor Minot and has been adopted only after trial of several other methods.
LDA summary

- LDA is a powerful model for
  - Visualizing the hidden thematic structure in large corpora
  - Generalizing new data to fit into that structure

- Builds on Deerwester et al. (1990) and Hofmann (1999)

- The main difference—LDA provides a generative probabilistic model of observed and future documents.
There is a growing research literature on topic models using LDA.

1. Importance of exploring data with unsupervised learning
2. Easy to encode assumptions with probabilistic models
3. Algorithmic advances in inference for large data sets
A variety of topic models
Dynamic topic models

Topics drifting in time
Analyzing a document

Original article

Sequencing the Genome, Fast
James C. Mabey and Andrew A. McPherson

..."
Sequencing the Genome, Fast

James C. Mastron and Andrew A. McPharland

Techeview: DNA Sequencing

Analyzing a document

sequence
genome
genes
sequences
human
gene
dna
sequencing
chromosome
regions
analysis
data
.genomic
number
devices
device
materials
.current
high
gate
light
si...
Visualizing trends within a topic

"Theoretical Physics"

"Neuroscience"
Correlated topic models

\[
\begin{align*}
\eta_d & \rightarrow Z_{d,n} \rightarrow W_{d,n} & \text{Noconjugate prior on topic proportions} \\
\Sigma & \rightarrow \eta_d & \beta_k \\
\mu & \rightarrow \Sigma
\end{align*}
\]

(See Blei and Lafferty, 2007b)
Ideal point topic models

Finds topics that can predict legislative votes from bill texts (93%)
Relational topic models

Finds topics in a **network** of documents (See Chang and Blei, 2010)
Still other ways to build on LDA

**New applications**—

- Syntactic topic models (Boyd-Graber and Blei 2009)
- Topic models on images (Fei-fei and Perona 2005 and others)
- Topic models on social network data (Airoldi et al. 2008)
- Topic models on music data (Hoffman et al. 2008)
- Modeling scientific impact (Gerrish and Blei 2010)
- Topic models for prediction (McAuliffe and Blei, 2007)

**Testing and relaxing assumptions**—

- “Infinite” numbers of topics (Teh et al., 2006)
- Spike and slab priors (Wang and Blei 2009 and Williamson et al. 2010)
- Models of word contagion (Elkan 2006)
- N-gram topic models (Wallach 2006)
- Human studies of interpretation (Chang et al. 2009)
Probabilistic modeling

Topic modeling is a part of the larger field of **probabilistic modeling**.

1. Treat data as observations that arise from a generative probabilistic process that includes **hidden variables**.
   - Here, the hidden variables represent the topic structure.

2. Infer the hidden structure using **posterior inference**.
   - What are the topics that describe my documents?

3. Situate **new data** into the estimated model.
   - How does this new document fit into the discovered topics?
There are many available implementations of topic modeling—

<table>
<thead>
<tr>
<th>Implementation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LDA-C</td>
<td>A C implementation of LDA</td>
</tr>
<tr>
<td>HDP</td>
<td>A C implementation of the HDP (“infinite LDA”)</td>
</tr>
<tr>
<td>Online LDA*</td>
<td>A python package for LDA on massive data</td>
</tr>
<tr>
<td>LDA in R*</td>
<td>Package in R for many topic models</td>
</tr>
<tr>
<td>LingPipe</td>
<td>Java toolkit for NLP and computational linguistics</td>
</tr>
<tr>
<td>Mallet</td>
<td>Java toolkit for statistical NLP</td>
</tr>
<tr>
<td>Soon...</td>
<td>A python package to build browsers from topic models</td>
</tr>
</tbody>
</table>

Example: LDA in R

docs <- read.documents("mult.dat")
K <- 20
alpha <- 1/20
eta <- 0.001
model <- lda.collapsed.gibbs.sampler(documents, K, vocab, 1000, alpha, eta)
Scalable topic modeling
The need for online inference

These topic models work in the same way—

- Posit a generative model.
- Cast the task at hand as a posterior computation
- Approximate the posterior.
The need for online inference

- Existing algorithms process document collections in **batch**.
  - Analyze every document.
  - Update the posterior.
  - Repeat.

- This is inefficient for large collections.

- It cannot accommodate streaming collections.
• We have developed an **online inference algorithm**
  • Obtain a document from a source.
  • Analyze that document.
  • Update the posterior.
  • Repeat.

• Allows us to analyze millions (even billions) of documents

• Lets us develop topic models on streaming collections
• Suppose (for the moment) the topics $\beta_{1:K}$ are fixed.
• For a document, computing the “local” posterior is intractable:

$$p(\theta | \alpha) \prod_{n=1}^{N} p(z_n | \theta) p(w_n | z_n, \beta_{1:K})$$

$$\int_\theta p(\theta | \alpha) \prod_{n=1}^{N} \sum_{z=1}^{K} p(z_n | \theta) p(w_n | z_n, \beta_{1:K})$$

• We will use variational inference to approximate it.
• (This will provides the right ingredients for online inference.)
Variational inference (in general)

- Let $x = x_{1:N}$ be observed variables; let $z = z_{1:M}$ be the latent variables.

- Our goal is to compute the posterior distribution

$$p(z \mid x) = \frac{p(z, x)}{\int p(z, x) \, dz}$$

- For many interesting distributions, the marginal likelihood of the observations is difficult to efficiently compute.
Variational inference

- Introduce a distribution over the latent variables $q_\nu(z)$, parameterized by variational parameters $\nu$.

- Use Jensen’s inequality to bound the log probability of the observations.

$$
\log p(x) = \log \int p(z, x) dz
\geq \log \int p(z, x) \frac{q_\nu(z)}{q_\nu(z)} dz
= \log \int p(z, x) q_\nu(z) dz
- E_{q_\nu} [\log p(Z, x)] - E_{q_\nu} [\log q_\nu(Z)]
$$

- We optimize the variational parameters to tighten this bound.

- This is the same as finding the member of the family $q_\nu$ that is closest in KL divergence to $p(z \mid x)$. 
Mean-field variational inference

- Complexity is determined by the factorization of $q_\nu$
- In mean field variational inference $q_\nu$ is fully factored
  \[ q_\nu(z) = \prod_{m=1}^{M} q_{\nu_m}(z_m). \]
- Each latent variable is independently governed by its own variational parameter $\nu_m$.
- In the true posterior they can exhibit dependence. (Often, this is what makes exact inference difficult.)
MFVI and conditional exponential families

- Assume that the joint distribution is in an exponential family,

\[ p(x, z \mid \eta) = \exp\{\eta^\top t(x, z) - a(\eta)\}. \]

- These assumptions describe many models.

- They imply that the conditional distribution of each latent variable given the observations and other latent variables is in an exponential family:

\[ p(z_m \mid z_{-m}, x) = h_m(z_m) \exp\{\eta(z_{-m}, x)^\top z_m - a_m(\eta(z_{-m}, x))\}. \]

The parameter is a function of the conditioned variables.
• Assume $q_\nu$ is fully factorized, and each factor is in the same exponential family as the corresponding conditional—

$$q_{\nu m}(z_m) = h_m(z_m) \exp\{\nu_m^T z_m - a_m(\nu_m)\}$$

• The evidence lower bound (ELBO) is

$$\log p(x) \geq E[\log p(x, Z)] - \sum_{m=1}^M E[\log q_m(Z|\nu_m)] = \mathcal{L}(\nu).$$

• The derivative with respect to each variational parameter is

$$\frac{\partial}{\partial \nu_m} \mathcal{L}(\nu) = a''(\nu_m)(E[\eta(Z_{-m}, x)] - \nu_m)$$
MFVI and conditional exponential families

- Variational inference is the following coordinate ascent algorithm
  \[ \nu_m = \mathbb{E}_{q_\nu}[\eta_m(Z_{-m}, x)] \]

- **Gibbs sampling**—
  Iteratively sample from the conditional distribution.

- **Variational inference**—
  Iteratively set each variational parameter equal to the expectation of the conditional natural parameter.

- (The mean field assumption makes this easy.)
We use the mean field variational distribution,

\[ q(\theta, z_{1:N} \mid \gamma, \phi_{1:N}) = q(\theta \mid \gamma) \prod_{n=1}^{N} q(z_n \mid \phi_n). \]

- Each variable is given its own parameter.
  - \( N \) topic assignment parameters \( \phi_n \)
  - One topic proportions parameter \( \gamma \)

- We optimize these parameters so \( q \) is close to the posterior.
Variational inference (topics fixed)

The coordinate ascent algorithm is—

\[
\gamma^{(t+1)} = \alpha + \sum_{n=1}^{N} \phi_n^{(t)}
\]

\[
\phi_n^{(t+1)} \propto \exp\{E[\log \theta] + \log \beta_{.,w_n}\},
\]

where

\[
E[\log \theta_i] = \Psi(\gamma_i) - \Psi(\sum_j \gamma_j).
\]

(For details, see Blei et al., 2003.)
**Example inference (again)**

## 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.
Estimating the topics

- We would like to estimate the topics as well.
- Construct a variational distribution for the whole collection.
  - $N \times D$ topic assignment parameters $\phi_{d,n}$
  - $D$ topic proportions parameters $\gamma_d$
  - $K$ topic parameters $\lambda$
- First do inference on each document. Then update topics,
  \[ \lambda_k = \eta + \sum_d \sum_n w_{d,n} \phi_{d,n} \]
Batch Variational Inference for LDA

1: Initialize topics $\lambda_{1:K}$ randomly.
2: while relative improvement in $\mathcal{L}(w, \phi, \gamma, \lambda) > \epsilon$ do
3: \hspace{1em} for $d = 1$ to $D$ do
4: \hspace{2em} Initialize $\gamma_{d,k} = 1$.
5: \hspace{2em} repeat
6: \hspace{3em} Set $\phi_{d,n} \propto \exp\{\mathbb{E}_q[\log \theta_d] + \mathbb{E}_q[\log \beta_{.w_n}]\}$
7: \hspace{3em} Set $\gamma_d = \alpha + \sum_n \phi_{d,n}$
8: \hspace{3em} until $\frac{1}{K} \sum_k |\text{change in } \gamma_{d,k}| < \epsilon$
9: \hspace{2em} end for
10: Set $\lambda_k = \eta + \sum_d \sum_n w_{d,n} \phi_{d,n}$
11: end while
Example topic inference

- human genome
dna genetic genes sequence gene molecular sequencing map information genetics mapping project sequences

- evolution evolutionary species organisms life origin biology groups phylogenetic living diversity group new two common
disease host bacteria diseases resistance bacterial new strains control infectious malaria parasite parasites united tuberculosis

- computer models information data computers system network systems model parallel methods networks software new simulations
1: Initialize topics \( \lambda_{1:K} \) randomly.
2: while relative improvement in \( \mathcal{L}(w, \phi, \gamma, \lambda) > \epsilon \) do
3:     for \( d = 1 \) to \( D \) do
4:         Initialize \( \gamma_{d,k} = 1 \).
5:     repeat
6:         Set \( \phi_{d,n} \propto \exp\{\mathbb{E}_q[\log \theta_d] + \mathbb{E}_q[\log \beta_{.,wn}]\} \)
7:         Set \( \gamma_d = \alpha + \sum_n \phi_{d,n} \)
8:     until \( \frac{1}{K} \sum_k |\text{change in } \gamma_{d,k}| < \epsilon \)
9: end for
10: Set \( \lambda_k = \eta + \sum_d \sum_n w_{d,n} \phi_{d,n} \)
11: end while

Do variational inference for each document.
“M step”

1: Initialize topics $\lambda_{1:K}$ randomly.
2: while relative improvement in $\mathcal{L}(w, \phi, \gamma, \lambda) > \epsilon$ do
3:     for $d = 1$ to $D$ do
4:         Initialize $\gamma_{d,k} = 1$.
5:         repeat
6:             Set $\phi_{d,n} \propto \exp\{\mathbb{E}_q[\log \theta_d] + \mathbb{E}_q[\log \beta, w_n]\}$
7:             Set $\gamma_d = \alpha + \sum_n \phi_{d,n}$
8:         until $\frac{1}{K} \sum_k | \text{change in } \gamma_{d,k} | < \epsilon$
9:     end for
10:    Set $\lambda_k = \eta + \sum_d \sum_n w_{d,n} \phi_{d,n}$
11: end while

Update the posterior estimates of the topics based on the “E step.”
The need for online inference

- Our goal is to use this (and related) models for analyzing massive collections of millions of documents.

- But, in the first step of batch inference we estimate the posterior for every document based on randomly initialized topics.

- Rather, we use **stochastic natural gradient ascent**.

- The basic procedure
  - Sample a document from a source
  - Process that document and update the model
  - Repeat
Why waste time with the real gradient, when a cheaper noisy estimate of the gradient will do (Robbins and Monro, 1951)?

Idea: Follow a noisy estimate of the gradient with a step-size.

By decreasing the step-size according to a certain schedule, we guarantee convergence to a local optimum.
Many models contain “local” and “global” variables.

- Local variables are drawn for each data point.
- Global variables are drawn once for the whole data set.

Idea—

- Subsample a small subset of the data
- Do variational inference for the local parameters
- Do stochastic optimization for the global parameters
Constructing a noisy estimate of the ELBO

- Construct an ELBO term for each set of local parameters—

\[ \mathcal{L}_d(\gamma_d, \phi_d, \lambda) = E[\log p(w_d, z_d, \theta_d \mid \beta)] + H(q(z_d, \theta_d)) \]

\[ + \frac{1}{D} (E[\log p(\beta)] + H(q(\beta))) \]

- The corpus ELBO is a sum of these terms, evaluated at the best local parameters—

\[ \mathcal{L}(\lambda) = \sum_{d=1}^{D} \mathcal{L}_d(\hat{\gamma}_d, \hat{\phi}_d, \lambda) \]

- Define a random variable \( I \sim \text{Unif}(1, \ldots D) \),

\[ E[D\mathcal{L}_I] = \mathcal{L} \]
The procedure is to—

- Draw an index $i$ at random
- Perform inference on document $i$ and the current topics
- Update the (global) topics with stochastic optimization

- No need to process the whole corpus before updating the model.
- Further, no need to keep the corpus around on disk!
Define \( \rho_t \triangleq (\tau_0 + t)^{-\kappa} \)

Initialize \( \lambda \) randomly.

**for** \( t = 0 \) to \( \infty \) **do**

Choose a random document \( w_t \)

Initialize \( \gamma_{tk} = 1 \). (The constant 1 is arbitrary.)

**repeat**

Set \( \phi_{t,n} \propto \exp\{\mathbb{E}_q[\log \theta_t] + \mathbb{E}_q[\log \beta_{..}w_n]\} \)

Set \( \gamma_t = \alpha + \sum_n \phi_{t,n} \)

**until** \( \frac{1}{K} \sum_k |\text{change in } \gamma_{t,k}| < \epsilon \)

Compute \( \lambda_k = \eta + D \sum_n w_{t,n} \phi_{t,n} \)

Set \( \lambda_k = (1 - \rho_t)\lambda_k + \rho_t \lambda_k \).

**end for**
Define $\rho_t \triangleq (\tau_0 + t)^{-\kappa}$

Initialize $\lambda$ randomly.

for $t = 0$ to $\infty$ do

Choose a random document $w_t$

Initialize $\gamma_{tk} = 1$. (The constant 1 is arbitrary.)

repeat

Set $\phi_{t,n} \propto \exp\{\mathbb{E}_q[\log \theta_t] + \mathbb{E}_q[\log \beta, w_n]\}$

Set $\gamma_t = \alpha + \sum_n \phi_{t,n}$

until $\frac{1}{K} \sum_k |\text{change in } \gamma_{t,k}| < \epsilon$

Compute $\tilde{\lambda}_k = \eta + D \sum_n w_{t,n} \phi_{t,n}$

Set $\lambda_k = (1 - \rho_t)\lambda_k + \rho_t \tilde{\lambda}_k$.

end for

The E-step only processes a single document.
Define $\rho_t \triangleq (\tau_0 + t)^{-\kappa}$

Initialize $\lambda$ randomly.

for $t = 0$ to $\infty$ do

Choose a random document $w_t$

Initialize $\gamma_{tk} = 1$. (The constant 1 is arbitrary.)

repeat

Set $\phi_{t,n} \propto \exp\{E_q[\log \theta_t] + E_q[\log \beta_{.,w_n}]\}$

Set $\gamma_t = \alpha + \sum_n \phi_{t,n}$

until $\frac{1}{K} \sum_k |\text{change in } \gamma_{t,k}| < \epsilon$

Compute $\tilde{\lambda}_k = \eta + D \sum_n w_{t,n} \phi_{t,n}$

Set $\lambda_k = (1 - \rho_t) \lambda_k + \rho_t \tilde{\lambda}_k$.

end for

The M-step treats that document as the whole corpus.
But, it only slightly adjusts the topics based on it.
Analyzing 3.3M articles from Wikipedia

**Documents analyzed**

<table>
<thead>
<tr>
<th>Documents analyzed</th>
<th>2048</th>
<th>4096</th>
<th>8192</th>
<th>12288</th>
<th>16384</th>
<th>32768</th>
<th>49152</th>
<th>65536</th>
</tr>
</thead>
<tbody>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>business</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
<tr>
<td>systems</td>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>business</td>
<td>systems</td>
<td>business</td>
</tr>
<tr>
<td>road</td>
<td>made</td>
<td>systems</td>
<td>service</td>
<td>companies</td>
<td>service</td>
<td>systems</td>
<td>companies</td>
<td>companies</td>
</tr>
</tbody>
</table>

**Top eight words**

- systems
- road
- made
- service
- announced
- national
- west
- language

**Perplexity**

- Online 98K
- Online 3.3M
- Batch 98K

**Documents seen (log scale)**

- $10^{3.5}$
- $10^4$
- $10^{4.5}$
- $10^5$
- $10^{5.5}$
- $10^6$
- $10^{6.5}$
• We can build topic models from streams of documents.

• Some issues
  • How quickly to decrease the learning rate?
  • What is the “batch size”?
  • How to interpret $D$ when there is a stream of documents?

• See the paper by Hoffman et al. (NIPS, 2010) and the foundational related work of Sato (Neural Computation, 2001).
Online inference is promising

- Stochastic variational methods are a general way to approximate the posterior for massive/streaming data.

- It is powerful for topic modeling, and can be adapted hierarchical models for many types of data:
  - E.g., neuroscience, biology, computer vision, social networks, social sciences, signal processing, ...
Summary and open directions
Hierarchical Bayesian models of text are a powerful way to explore, summarize and search large archives of documents.

Generally, probabilistic models provide an intuitive language for expressing assumptions about observed and hidden variables.

Algorithmic advances in approximate posterior inference let us apply complex models to large real-world data sets.
Open research directions

- **Model diagnostics and model checking**
  Which model should I choose for which task?

- **Incorporating corpus, discourse, or linguistic structure**
  How can our knowledge of language help us build and use exploratory models of text?

- **Interfaces and “downstream” applications of topic modeling**
  What can I do with an annotated corpus?

- **Theoretical understanding of approximate inference**
  What do we know about variational inference from either the statistical or learning perspective?
“We should seek out unfamiliar summaries of observational material, and establish their useful properties... And still more novelty can come from finding, and evading, still deeper lying constraints.” (J. Tukey, *The Future of Data Analysis*, 1962)
Ideal point topic models
The ideal point model

- A model devised to uncover voting patterns (Clinton et al., 2004).
- We observe roll call data $v_{ij}$.
- Bills attached to discrimination parameters $a_j$.
- Senators attached to ideal points $x_i$.

\[ p(v_{ij}) = f(d(x_i, a_j)) \]
The ideal point model

- Posterior inference reveals the political spectrum of senators
- Widely used in quantitative political science.
The ideal point model is limited for prediction

\[ p(v_{ij}) = f(d(x_i, a_j)) \]

- We can predict a missing vote.
- But we cannot predict all the missing votes from a bill.
- Cf. the limitations of collaborative filtering
Ideal point topic models

Use supervised topic modeling assumptions as a predictive mechanism from bill texts to bill discrimination.
Ideal point topic models
In addition to senators and bills, IPTM places **topics** on the spectrum.
Versus the LASSO, the IPTM correctly predicted 126,000 more votes.
Ideal point topic models

- Ideal point topic model illustrates
  - Topic modeling embedded in a complex model
  - Topic modeling used to solve a real-world problem with text

- More generally, consider collaborative filtering.
  - Senators are *users*.
  - Bills are *items*.

- Existing collaborative filtering is akin to classical ideal point.

- Our model lets us predict preferences on *completely new items.*