Unsupervised Clustering

Digging into Data: Jordan Boyd-Graber

University of Maryland

April 8, 2013

COLLEGE OF INFORMATION STUDIES

Outline

1. Topic Model Introduction

Why topic models?

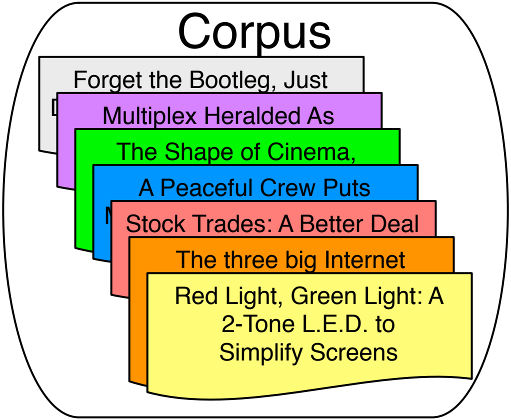
- Suppose you have a huge number of documents
- Want to know what's going on
- Can't read them all (e.g. every New York Times article from the 90's)
- Topic models offer a way to get a corpus-level view of major themes

Why topic models?

- Suppose you have a huge number of documents
- Want to know what's going on
- Can't read them all (e.g. every New York Times article from the 90's)
- Topic models offer a way to get a corpus-level view of major themes
- Unsupervised

Conceptual Approach

From an **input corpus** and number of topics K → words to topics



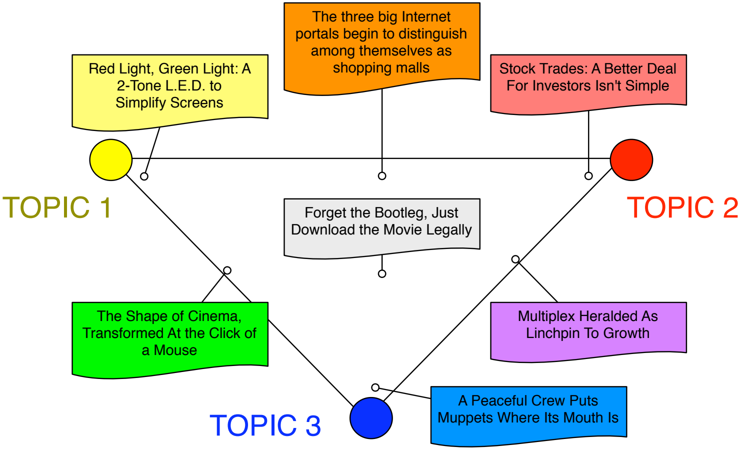
Conceptual Approach

From an input corpus and number of topics K $\longrightarrow$ **words to topics**

TOPIC 1	TOPIC 2	TOPIC 3
computer, technology, system, service, site, phone, internet, machine	sell, sale, store, product, business, advertising, market, consumer	play, film, movie, theater, production, star, director, stage

Conceptual Approach

- For each document, what topics are expressed by that document?



Topics from *Science*

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

Why should you care?

- Neat way to explore / understand corpus collections
  - E-discovery
  - Social media
  - Scientific data
- NLP Applications
  - POS Tagging [9]
  - Word Sense Disambiguation [2]
  - Word Sense Induction [3]
  - Discourse Segmentation [8]
- Psychology [5]: word meaning, polysemy
- Inference is (relatively) simple

Matrix Factorization Approach

$$\underbrace{\begin{bmatrix} M \times K \end{bmatrix}}_{\text{Topic Assignment}} \times \underbrace{\begin{bmatrix} \quad K \times V \quad \end{bmatrix}}_{\text{Topics}} \approx \underbrace{\begin{bmatrix} \quad M \times V \quad \end{bmatrix}}_{\text{Dataset}}$$

- **K** Number of topics
- **M** Number of documents
- **V** Size of vocabulary

Matrix Factorization Approach

$$\underbrace{\begin{bmatrix} M \times K \end{bmatrix}}_{\text{Topic Assignment}} \times \underbrace{\begin{bmatrix} K \times V \end{bmatrix}}_{\text{Topics}} \approx \underbrace{\begin{bmatrix} M \times V \end{bmatrix}}_{\text{Dataset}}$$

K Number of topics
M Number of documents
V Size of vocabulary

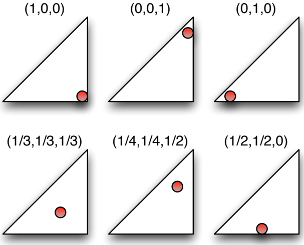
- If you use singular value decomposition (SVD), this technique is called latent semantic analysis.
- Popular in information retrieval.

Alternative: Generative Model

- How your data came to be
- Sequence of Probabilistic Steps
- Posterior Inference

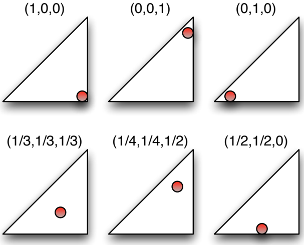
Multinomial Distribution

- Distribution over discrete outcomes
- Represented by non-negative vector that sums to one
- Picture representation



Multinomial Distribution

- Distribution over discrete outcomes
- Represented by non-negative vector that sums to one
- Picture representation



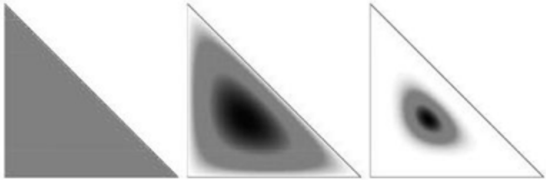
- Come from a Dirichlet distribution

Dirichlet Distribution

$$P(\boldsymbol{p} \,|\, \alpha \boldsymbol{m}) = \frac{\Gamma(\sum_k \alpha m_k)}{\prod_k \Gamma(\alpha m_k)} \prod_k p_k^{\alpha m_k - 1}$$

Dirichlet Distribution

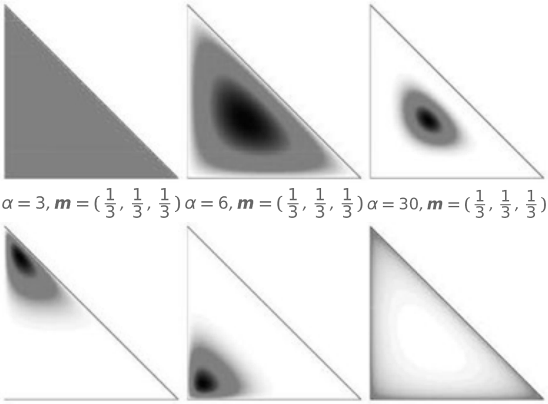
$$P(\boldsymbol{p} \mid \alpha \boldsymbol{m}) = \frac{\Gamma(\sum_k \alpha m_k)}{\prod_k \Gamma(\alpha m_k)} \prod_k p_k^{\alpha m_k - 1}$$



$\alpha = 3, \boldsymbol{m} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$ $\alpha = 6, \boldsymbol{m} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$ $\alpha = 30, \boldsymbol{m} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$

Dirichlet Distribution

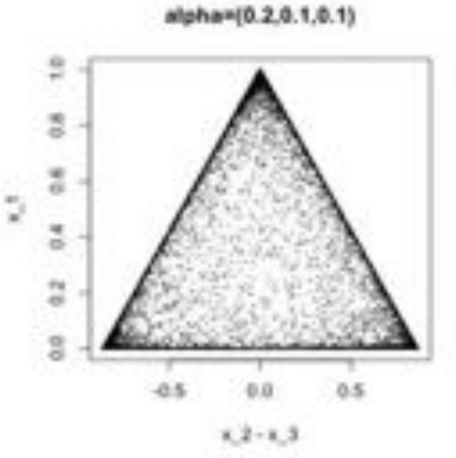
$$P(\boldsymbol{p} \mid \alpha \boldsymbol{m}) = \frac{\Gamma(\sum_k \alpha m_k)}{\prod_k \Gamma(\alpha m_k)} \prod_k p_k^{\alpha m_k - 1}$$



$\alpha = 3, \boldsymbol{m} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$ $\alpha = 6, \boldsymbol{m} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$ $\alpha = 30, \boldsymbol{m} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$

$\alpha = 14, \boldsymbol{m} = (\frac{1}{7}, \frac{5}{7}, \frac{1}{7})$ $\alpha = 14, \boldsymbol{m} = (\frac{1}{7}, \frac{1}{7}, \frac{5}{7})$ $\alpha = 2.7, \boldsymbol{m} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$

Dirichlet Distribution



Dirichlet Distribution

- If $\phi \sim \text{Dir}(()\alpha)$, $\boldsymbol{w} \sim \text{Mult}(()\phi)$, and $n_k = |\{w_i : w_i = k\}|$ then

$$
\begin{aligned}
p(\phi|\alpha,\boldsymbol{w}) &\propto p(\boldsymbol{w}|\phi)p(\phi|\alpha) && (1)\\
&\propto \prod_k \phi^{n_k} \prod_k \phi^{\alpha_k - 1} && (2)\\
&\propto \prod_k \phi^{\alpha_k + n_k - 1} && (3)
\end{aligned}
$$

- Conjugacy: this **posterior** has the same form as the **prior**

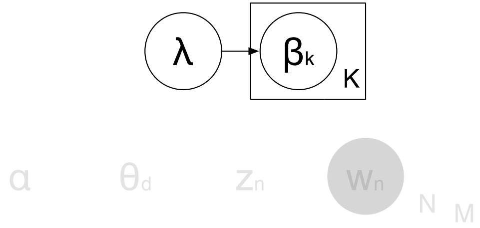
Dirichlet Distribution

- If $\phi \sim \text{Dir}(()\alpha)$, $\boldsymbol{w} \sim \text{Mult}(()\phi)$, and $n_k = |\{w_i : w_i = k\}|$ then

$$
\begin{aligned}
p(\phi|\alpha, \boldsymbol{w}) &\propto p(\boldsymbol{w}|\phi)p(\phi|\alpha) &(1)\\
&\propto \prod_k \phi^{n_k} \prod_k \phi^{\alpha_k - 1} &(2)\\
&\propto \prod_k \phi^{\alpha_k + n_k - 1} &(3)
\end{aligned}
$$

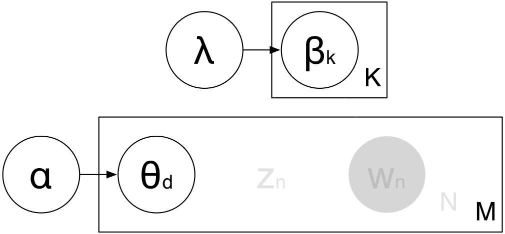
- Conjugacy: this **posterior** has the same form as the **prior**

Generative Model Approach



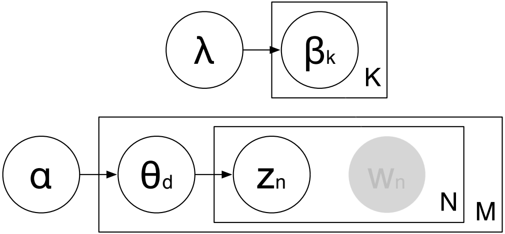
- For each topic $k \in \{1, \ldots, K\}$, draw a multinomial distribution β_k from a Dirichlet distribution with parameter λ

Generative Model Approach



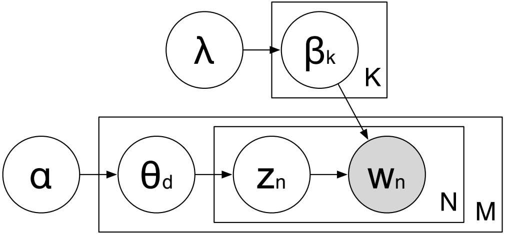
- For each topic $k \in \{1, \dots, K\}$, draw a multinomial distribution β_k from a Dirichlet distribution with parameter λ
- For each document $d \in \{1, \dots, M\}$, draw a multinomial distribution θ_d from a Dirichlet distribution with parameter α

Generative Model Approach



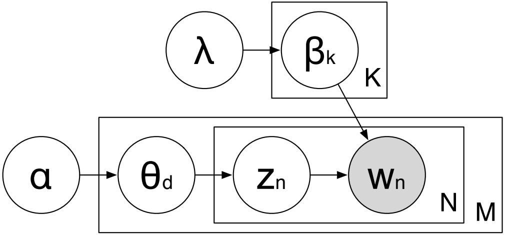
- For each topic $k \in \{1, \dots, K\}$, draw a multinomial distribution β_k from a Dirichlet distribution with parameter λ
- For each document $d \in \{1, \dots, M\}$, draw a multinomial distribution θ_d from a Dirichlet distribution with parameter α
- For each word position $n \in \{1, \dots, N\}$, select a hidden topic z_n from the multinomial distribution parameterized by θ.

Generative Model Approach



- For each topic $k \in \{1, \ldots, K\}$, draw a multinomial distribution β_k from a Dirichlet distribution with parameter λ
- For each document $d \in \{1, \ldots, M\}$, draw a multinomial distribution θ_d from a Dirichlet distribution with parameter α
- For each word position $n \in \{1, \ldots, N\}$, select a hidden topic z_n from the multinomial distribution parameterized by θ.
- Choose the observed word w_n from the distribution β_{z_n}.

Generative Model Approach



- For each topic $k \in \{1, \dots, K\}$, draw a multinomial distribution β_k from a Dirichlet distribution with parameter λ
- For each document $d \in \{1, \dots, M\}$, draw a multinomial distribution θ_d from a Dirichlet distribution with parameter α
- For each word position $n \in \{1, \dots, N\}$, select a hidden topic z_n from the multinomial distribution parameterized by θ.
- Choose the observed word w_n from the distribution β_{z_n}.

We use statistical inference to uncover the most likely unobserved variables given observed data.

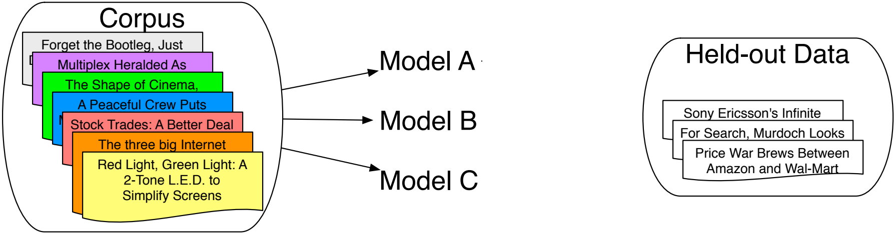
Topic Models: What's Important

- Topic models
  - Topics to words—multinomial distribution
  - Documents to topics—multinomial distribution
- Focus in this talk: statistical methods
  - Model: story of how your data came to be
  - Latent variables: missing pieces of your story
  - Statistical inference: filling in those missing pieces
- We use latent Dirichlet allocation (LDA) [1], a fully Bayesian version of pLSI [6], probabilistic version of LSA [7]

Topic Models: What's Important

- Topic models (latent variables)
  - Topics to words—multinomial distribution
  - Documents to topics—multinomial distribution
- Focus in this talk: statistical methods
  - Model: story of how your data came to be
  - Latent variables: missing pieces of your story
  - Statistical inference: filling in those missing pieces
- We use latent Dirichlet allocation (LDA) [1], a fully Bayesian version of pLSI [6], probabilistic version of LSA [7]

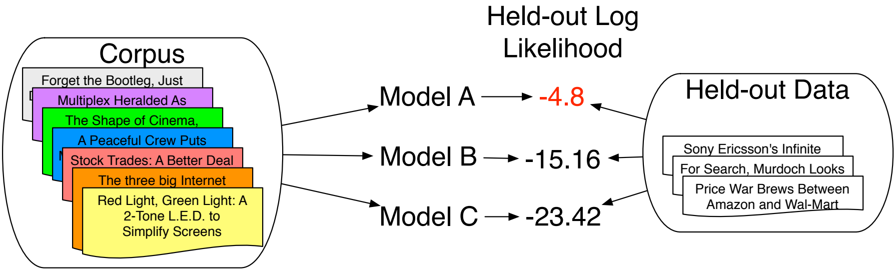
Evaluation



$$P(\mathbf{w} \mid \mathbf{w}', \mathbf{z}', \alpha\mathbf{m}, \beta\mathbf{u}) = \sum_{\mathbf{z}} P(\mathbf{w}, \mathbf{z} \mid \mathbf{w}', \mathbf{z}', \alpha\mathbf{m}, \beta\mathbf{u})$$

How you compute it is important too [10]

Evaluation



Measures predictive power, not what the topics are

$$P(\mathbf{w} \mid \mathbf{w}', \mathbf{z}', \alpha \mathbf{m}, \beta \mathbf{u}) = \sum_{\mathbf{z}} P(\mathbf{w}, \mathbf{z} \mid \mathbf{w}', \mathbf{z}', \alpha \mathbf{m}, \beta \mathbf{u})$$

How you compute it is important too [10]

Word Intrusion

TOPIC 1	TOPIC 2	TOPIC 3
computer, technology, system, service, site, phone, internet, machine	sell, sale, store, product, business, advertising, market, consumer	play, film, movie, theater, production, star, director, stage

Word Intrusion

1. Take the highest probability words from a topic

Original Topic

dog, cat, horse, pig, cow

Word Intrusion

1. Take the highest probability words from a topic

Original Topic

dog, cat, horse, pig, cow

2. Take a high-probability word from another topic and add it

Topic with Intruder

dog, cat, apple, horse, pig, cow

Word Intrusion

1. Take the highest probability words from a topic

Original Topic

dog, cat, horse, pig, cow

2. Take a high-probability word from another topic and add it

Topic with Intruder

dog, cat, apple, horse, pig, cow

3. We ask users to find the word that doesn't belong

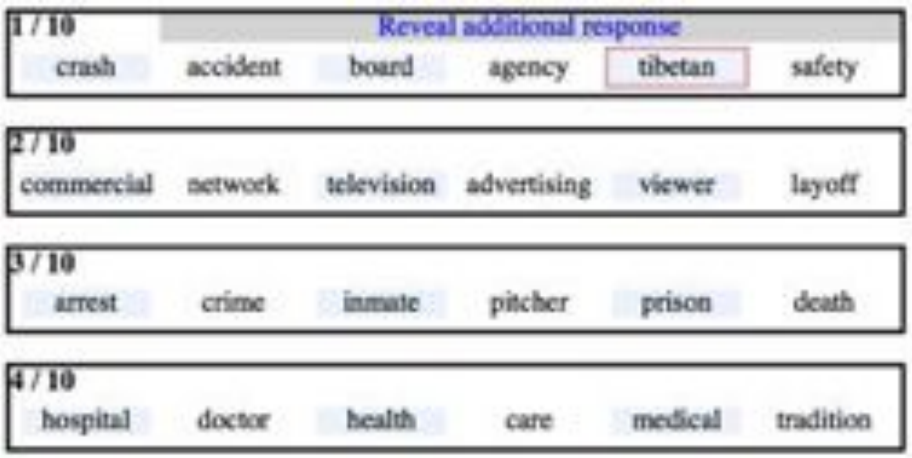
Hypothesis

If the topics are interpretable, users will consistently choose true intruder

Word Intrusion

1 / 10	crash	accident	board	agency	tibetan	safety
2 / 10	commercial	network	television	advertising	viewer	layoff
3 / 10	arrest	crime	inmate	pitcher	prison	death
4 / 10	hospital	doctor	health	care	medical	tradition

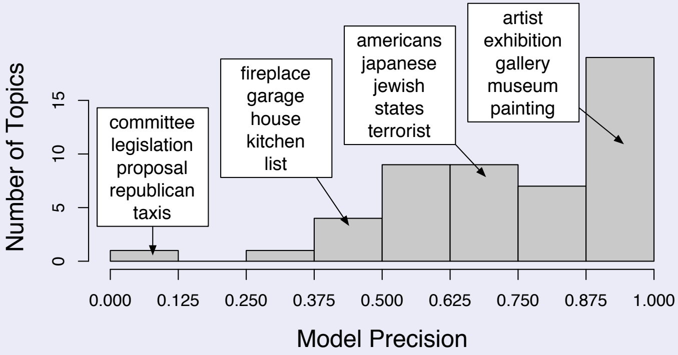
Word Intrusion



- Order of words was shuffled
- Which intruder was selected varied
- Model precision: percentage of users who clicked on intruder

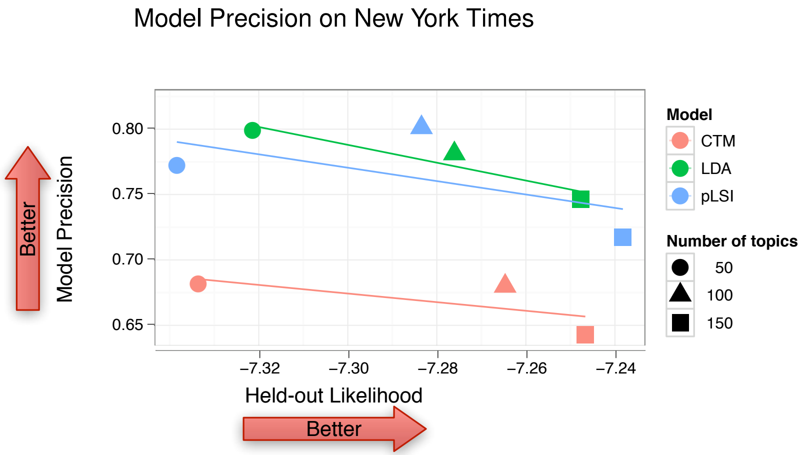
Word Intrusion: Which Topics are Interpretable?

New York Times, 50 LDA Topics



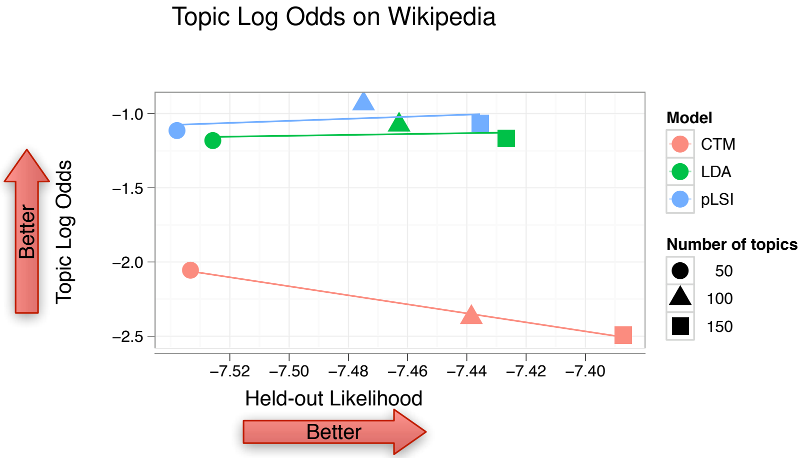
Model Precision: percentage of correct intruders found

Interpretability and Likelihood



within a model, higher likelihood $\neq$ higher interpretability

Interpretability and Likelihood



across models, higher likelihood $\neq$ higher interpretability

Evaluation Takeaway

- Measure what you care about [4]
- If you care about prediction, likelihood is good
- If you care about a particular task, measure that

Gibbs Sampling

- A way to go from random topics (i.e., bad) to good topics (i.e., ones that make sense)
- We do this by changing the topic assignment of a word $z_{d,n}$
- Given a state $\{z_1, \dots z_N\}$, drawing $z_n \sim p(n_n | z_1, \dots z_{n-1}, z_{n+1}, \dots z_N, X, \Theta)$ for all n (repeatedly) results in the distribution of topics **given documents.**
- For notational convenience, call $\boldsymbol{z}$ with $z_{d,n}$ removed $\boldsymbol{z}_{-d,n}$

Inference

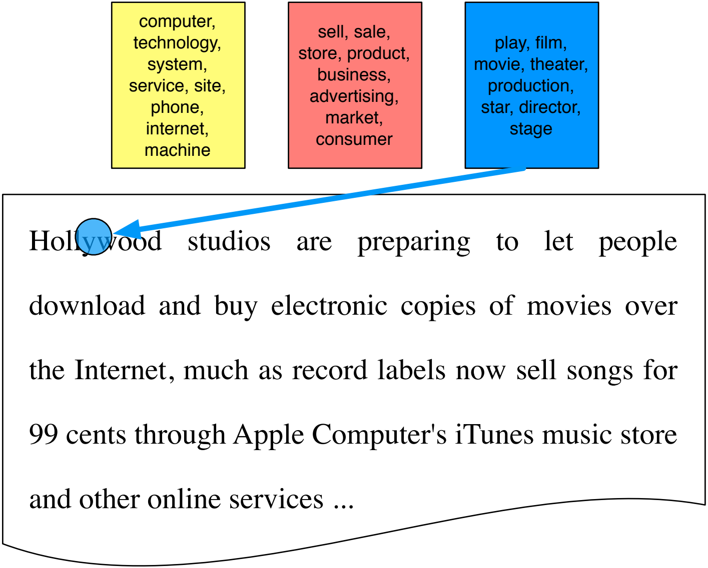
computer, technology, system, service, site, phone, internet, machine

sell, sale, store, product, business, advertising, market, consumer

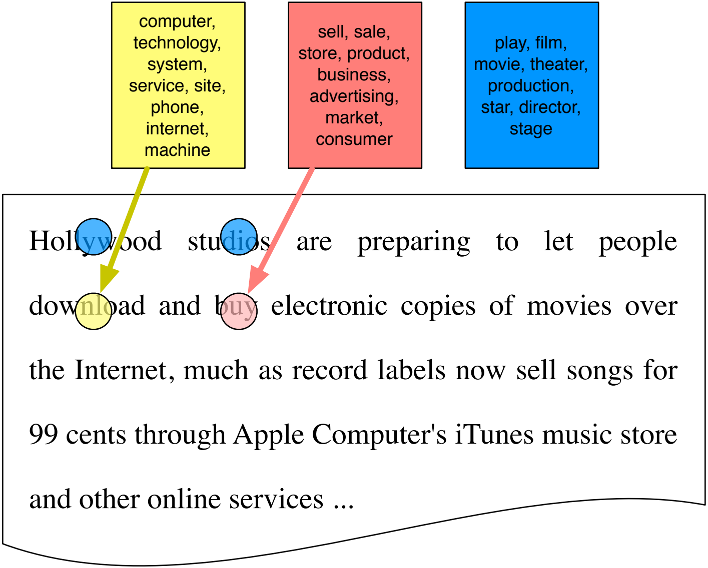
play, film, movie, theater, production, star, director, stage

Hollywood studios are preparing to let people download and buy electronic copies of movies over the Internet, much as record labels now sell songs for 99 cents through Apple Computer's iTunes music store and other online services ...

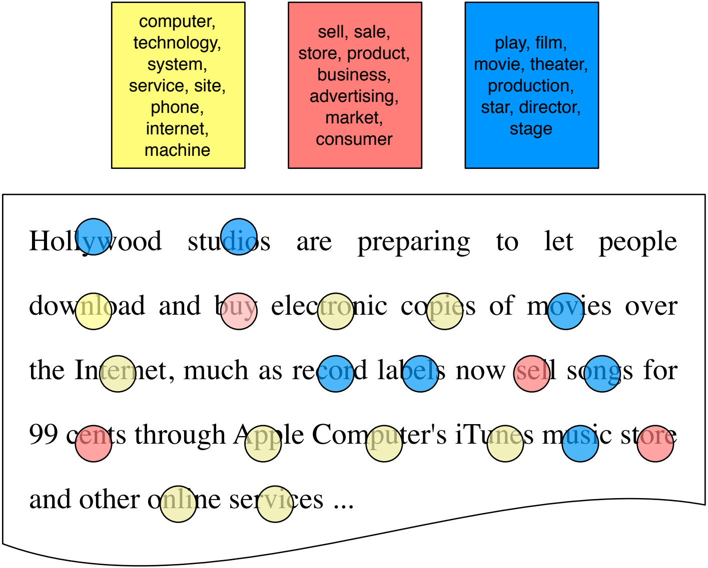
Inference


computer, technology, system, service, site, phone, internet, machine
sell, sale, store, product, business, advertising, market, consumer
play, film, movie, theater, production, star, director, stage
Hollywood studios are preparing to let people download and buy electronic copies of movies over the Internet, much as record labels now sell songs for 99 cents through Apple Computer's iTunes music store and other online services ...

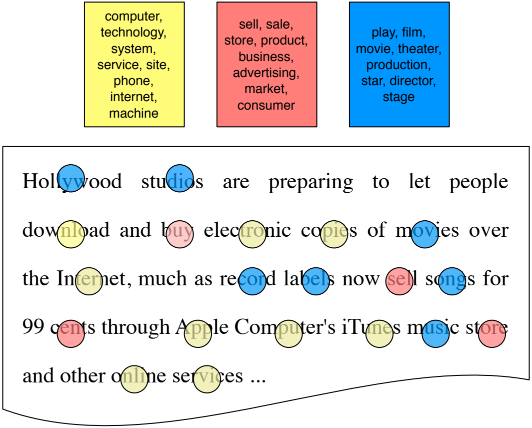
Inference


computer, technology, system, service, site, phone, internet, machine
sell, sale, store, product, business, advertising, market, consumer
play, film, movie, theater, production, star, director, stage
Hollywood studios are preparing to let people download and buy electronic copies of movies over the Internet, much as record labels now sell songs for 99 cents through Apple Computer's iTunes music store and other online services ...

Inference


computer, technology, system, service, site, phone, internet, machine
sell, sale, store, product, business, advertising, market, consumer
play, film, movie, theater, production, star, director, stage
Hollywood studios are preparing to let people download and buy electronic copies of movies over the Internet, much as record labels now sell songs for 99 cents through Apple Computer's iTunes music store and other online services ...

Inference



And repeat, conditioning $z_{d,n}$ on all of the other assignments

Gibbs Sampling

- For LDA, we will sample the topic assignments
- The topics and per-document topic proportions are integrated out / marginalized / Rao-Blackwellized
- Thus, we want:

$$p(z_{d,n} = k | \boldsymbol{z}_{-d,n}, \boldsymbol{w}, \alpha, \lambda) = \frac{p(z_{d,n} = k, \boldsymbol{z}_{-d,n} | \boldsymbol{w}, \alpha, \lambda)}{p(\boldsymbol{z}_{-d,n} | \boldsymbol{w}, \alpha, \lambda)}$$

Gibbs Sampling

- For LDA, we will sample the topic assignments
- The topics and per-document topic proportions are integrated out / marginalized / Rao-Blackwellized
- Thus, we want:

$$p(z_{d,n} = k | \boldsymbol{z}_{-d,n}, \boldsymbol{w}, \alpha, \lambda) = \frac{p(z_{d,n} = k, \boldsymbol{z}_{-d,n} | \boldsymbol{w}, \alpha, \lambda)}{p(\boldsymbol{z}_{-d,n} | \boldsymbol{w}, \alpha, \lambda)}$$

- Let $n_{d,i}$ be the number of words taking topic i in document d. Let $v_{k,w}$ be the number of times word w is used in topic k.

$$= \frac{n_{d,k} + \alpha_k}{\sum_i^K n_{d,i} + \alpha_i} \frac{v_{k,w_{d,n}} + \lambda_{w_{d,n}}}{\sum_i v_{k,i} + \lambda_i}$$

Sample Document

Etruscan	trade	price	temple	market

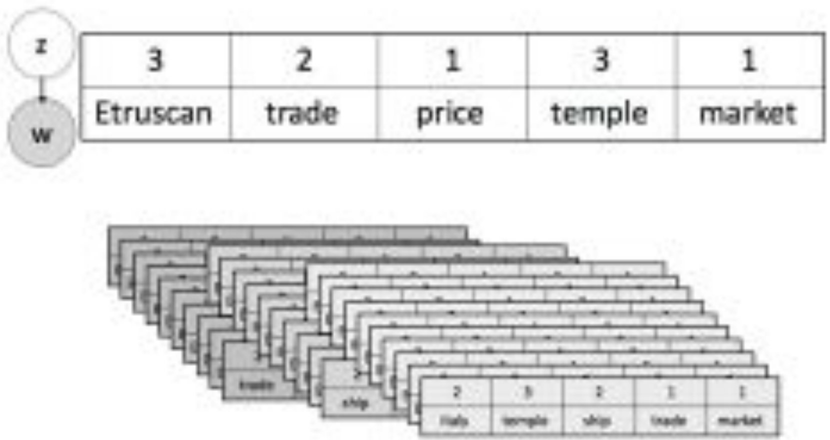
Sample Document

Etruscan	trade	price	temple	market

Randomly Assign Topics

3	2	1	3	1
Etruscan	trade	price	temple	market

Randomly Assign Topics



Total Topic Counts

3	2	1	3	1
Etruscan	trade	price	temple	market

Total counts from all docs

	1	2	3
Etruscan	1	0	35
market	50	0	1
price	42	1	0
temple	0	0	20
trade	10	8	1
...			

Total Topic Counts

3	2	1	3	1
Etruscan	trade	price	temple	market

Total counts from all

	1	2	3
Etruscan	1	0	35
market	50	0	1
price	42	1	0
temple	0	0	20

Sampling Equation

$$\frac{n_{d,k} + \alpha_k}{\sum_i^K n_{d,i} + \alpha_i} \frac{v_{k,w_{d,n}} + \lambda_{w_{d,n}}}{\sum_i v_{k,i} + \lambda_i}$$

Total Topic Counts

3	2	1	3	1
Etruscan	trade	price	temple	market

Total counts from all

	1	2	3
Etruscan	1	0	35
market	50	0	1
price	42	1	0
temple	0	0	20

Sampling Equation

$$\frac{n_{d,k} + \alpha_k}{\sum_i^K n_{d,i} + \alpha_i} \frac{v_{k,w_{d,n}} + \lambda_{w_{d,n}}}{\sum_i v_{k,i} + \lambda_i}$$

We want to sample this word . . .

3	**2**	1	3	1
Etruscan	trade	price	temple	market

	1	2	3
Etruscan	1	0	35
market	50	0	1
price	42	1	0
temple	0	0	20
trade	10	8	1
...			

We want to sample this word . . .

3	2	1	3	1
Etruscan	trade	price	temple	market

	1	2	3
Etruscan	1	0	35
market	50	0	1
price	42	1	0
temple	0	0	20
trade	10	8	1
...			

Decrement its count

3	?	1	3	1
Etruscan	trade	price	temple	market

	1	2	3
Etruscan	1	0	35
market	50	0	1
price	42	1	0
temple	0	0	20
trade	10	7	1
...			

What is the conditional distribution for this topic?

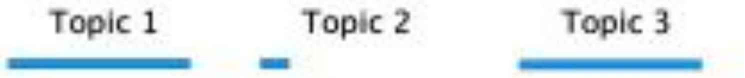
3	?	1	3	1
Etruscan	trade	price	temple	market

Part 1: How much does this document like each topic?

3	?	1	3	1
Etruscan	trade	price	temple	market

Part 1: How much does this document like each topic?

3	?	1	3	1
Etruscan	trade	price	temple	market



Part 1: How much does this document like each topic?

3	?	1	3	1
Etruscan	trade	price	temple	market

Topic 1 Topic 2 Topic 3

Sampling Equation

$$\frac{n_{d,k} + \alpha_k}{\sum_i^K n_{d,i} + \alpha_i} \frac{v_{k,w_{d,n}} + \lambda_{w_{d,n}}}{\sum_i v_{k,i} + \lambda_i}$$

Part 1: How much does this document like each topic?

3	?	1	3	1
Etruscan	trade	price	temple	market

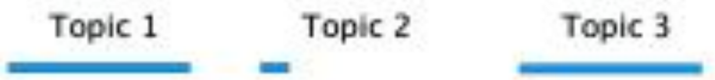
Topic 1 Topic 2 Topic 3

Sampling Equation

$$\frac{n_{d,k} + \alpha_k}{\sum_i^K n_{d,i} + \alpha_i} \frac{v_{k,w_{d,n}} + \lambda_{w_{d,n}}}{\sum_i v_{k,i} + \lambda_i}$$

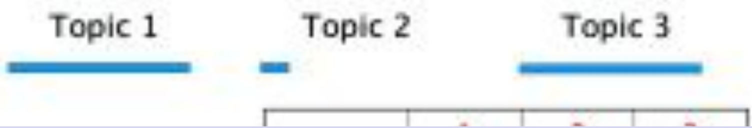
Part 2: How much does each topic like the word?

3	?	1	3	1
Etruscan	trade	price	temple	market



	1	2	3
trade	10	7	1

3	?	1	3	1
Etruscan	trade	price	temple	market

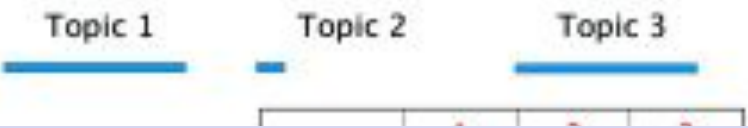


Sampling Equation

$$\frac{n_{d,k} + \alpha_k}{\sum_i^K n_{d,i} + \alpha_i} \frac{v_{k,w_{d,n}} + \lambda_{w_{d,n}}}{\sum_i v_{k,i} + \lambda_i}$$

Part 2: How much does each topic like the word?

3	?	1	3	1
Etruscan	trade	price	temple	market

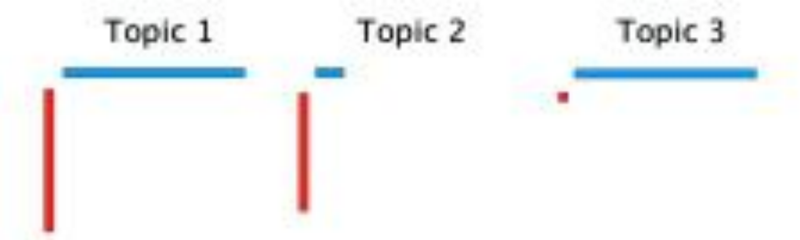


Sampling Equation

$$\frac{n_{d,k} + \alpha_k}{\sum_i^K n_{d,i} + \alpha_i} \frac{v_{k,w_{d,n}} + \lambda_{w_{d,n}}}{\sum_i v_{k,i} + \lambda_i}$$

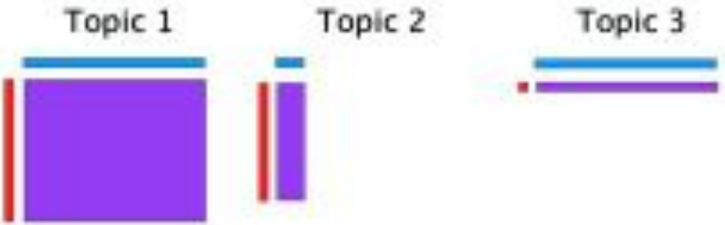
Geometric interpretation

3	?	1	3	1
Etruscan	trade	price	temple	market



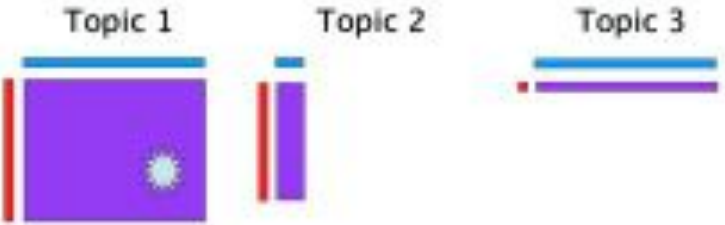
Geometric interpretation

3	?	1	3	1
Etruscan	trade	price	temple	market



Geometric interpretation

3	?	1	3	1
Etruscan	trade	price	temple	market



Update counts

3	?	1	3	1
Etruscan	trade	price	temple	market

	1	2	3
Etruscan	1	0	35
market	50	0	1
price	42	1	0
temple	0	0	20
trade	**10**	7	1
...			

Update counts

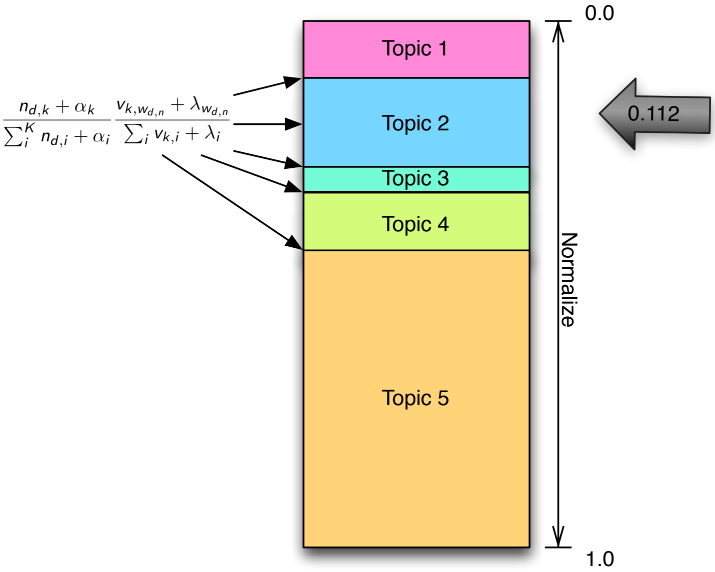
3	1	1	3	1
Etruscan	trade	price	temple	market

	1	2	3
Etruscan	1	0	35
market	50	0	1
price	42	1	0
temple	0	0	20
trade	11	7	1
...			

Update counts

3	1	1	3	1
Etruscan	trade	price	temple	market

Details: how to sample from a distribution



Algorithm

1. For each iteration i:
   1. For each document d and word n currently assigned to z_{old}:
      1. Decrement $n_{d,z_{old}}$ and $v_{z_{old},w_{d,n}}$
      2. Sample $z_{new} = k$ with probability proportional to $\frac{n_{d,k}+\alpha_k}{\sum_i^K n_{d,i}+\alpha_i} \frac{v_{k,w_{d,n}}+\lambda_{w_{d,n}}}{\sum_i v_{k,i}+\lambda_i}$
      3. Increment $n_{d,z_{new}}$ and $v_{z_{new},w_{d,n}}$

Implementation

Algorithm

1. For each iteration i:
   1. For each document d and word n currently assigned to z_{old}:
      1. Decrement $n_{d,z_{old}}$ and $v_{z_{old},w_{d,n}}$
      2. Sample $z_{new} = k$ with probability proportional to $\frac{n_{d,k}+\alpha_k}{\sum_i^K n_{d,i}+\alpha_i} \frac{v_{k,w_{d,n}}+\lambda_{w_{d,n}}}{\sum_i v_{k,i}+\lambda_i}$
      3. Increment $n_{d,z_{new}}$ and $v_{z_{new},w_{d,n}}$

Desiderata

- Hyperparameters: Sample them too (slice sampling)
- Initialization: Random
- Sampling: Until likelihood converges
- Lag / burn-in: Difference of opinion on this
- Number of chains: Should do more than one

Available implementations

- Mallet (http://mallet.cs.umass.edu)
- LDAC (http://www.cs.princeton.edu/ blei/lda-c)
- Topicmod (http://code.google.com/p/topicmod)
- LDA R pakcage (http://cran.r-project.org/web/packages/lda/lda.pdf)

[1] David M. Blei, Andrew Ng, and Michael Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[2] Jordan Boyd-Graber, David M. Blei, and Xiaojin Zhu. A topic model for word sense disambiguation. In *Proceedings of Emperical Methods in Natural Language Processing*, 2007.

[3] Samuel Brody and Mirella Lapata. Bayesian word sense induction. In *Proceedings of the European Chapter of the Association for Computational Linguistics*, Athens, Greece, 2009.

[4] Jonathan Chang, Jordan Boyd-Graber, and David M. Blei. Connections between the lines: Augmenting social networks with text. In *Knowledge Discovery and Data Mining*, 2009.

[5] Thomas L. Griffiths, Mark Steyvers, and Joshua Tenenbaum. Topics in semantic representation. *Psychological Review*, 114(2):211–244, 2007.

[6] Thomas Hofmann. Probabilistic latent semantic analysis. In *Proceedings of Uncertainty in Artificial Intelligence*, 1999.

[7] T. Landauer and S. Dumais. Solutions to Plato's problem: The latent semantic analsyis theory of acquisition, induction and representation of knowledge. *Psychological Review*, (104), 1997.

[8] Matthew Purver, Konrad Körding, Thomas L. Griffiths, and Joshua Tenenbaum. Unsupervised topic modelling for multi-party spoken discourse. In *Proceedings of the Association for Computational Linguistics*, 2006.

[9] Kristina Toutanova and Mark Johnson. A Bayesian LDA-based model for semi-supervised part-of-speech tagging. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 1521–1528. MIT Press, Cambridge, MA, 2008.

[10] Hanna Wallach, David Mimno, and Andrew McCallum. Rethinking LDA: Why priors matter. In *Proceedings of Advances in Neural Information Processing Systems*, 2009.