

CSC2535 Spring 2010
Advanced Machine Learning

Models of Text & Documents

Richard Zemel

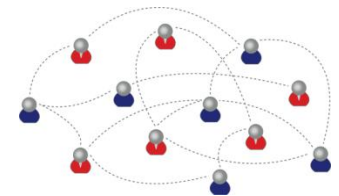
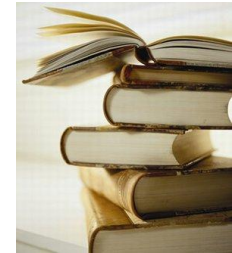
Outline

Models of words and documents

- Simple document models
- Probabilistic document models
 - Aspect model
 - Latent Dirichlet Allocation
- Extensions of topic models
 - Author-recipient topic model
 - Dynamic topic model
 - Hierarchical topic models
- Topic models and vision

Topic Models

- Have been applied to many types of data
 - Text
 - Images
 - Biological data
 - Relational data
 - Videos
 - and more...



Document Modeling

- automated analysis, visualization of text documents: crucial to effective use of large text archives (news stories, email collections, web)
- information retrieval: one of largest application areas of ML, growing steadily
- for example, next generation of web searching will likely rely on automated summarization; paper-reviewer matching example
- today: statistical models of documents and text; examples of influential/interesting models

Representations of Documents

standard document representation: count occurrences of each word stem (**bag-of-words**)

$$P(\{w_1, w_2, \dots, w_N\}) = \prod_{n=1}^N P(w_n)$$

The image shows a screenshot of a website page for TOTAL. The page has a header with the TOTAL logo and a navigation menu. The main content area is titled "all about the company" and contains several paragraphs of text. The text describes the company's energy exploration, production, and distribution operations, its strength in oil and gas reserves, and its expanding refining and marketing operations in Asia and the Mediterranean Rim. The page also mentions its growing specialty chemicals sector.

aardvark	0
about	2
all	2
Africa	1
apple	0
anxious	0
...	
gas	1
...	
oil	1
...	
Zaire	0

Representations of Documents

D documents; W distinct words \rightarrow

$F = W \times D$ word-count matrix

Does high value of f_{wd} indicate an important word?

One transform: tf-idf (term frequency-inverse document frequency) \rightarrow

$G = W \times D$ matrix of tf-idf values = tf * idf

$$\text{tf}_{wd} = P(w | d) = f_{wd} / \sum_{w'} f_{w'd} \quad \text{idf}_w = D / \sum_d [f_{wd} > 0]$$

Used to represent search query: sum of tf-idf of each query words

Topic Modeling

Aim: Find low-dimensional description of high-dimensional text

From ML point of view - just a latent variable problem!

Topic models facilitate:

- Summarization: find concise restatements
- Similarity: evaluate distance between texts

Latent Semantic Analysis/Indexing

Reduced representation of F : apply SVD

$$\begin{array}{c} \text{Words} \\ \boxed{F} \\ \text{Documents} \end{array} = \begin{array}{c} \boxed{A} \\ W \times M \end{array} \begin{array}{c} \boxed{D} \\ M \times M \end{array} \begin{array}{c} \boxed{B} \\ M \times D \end{array}$$

- reduced representation of word i : row of AD -- can describe semantic relationships
- relationships between words described by cosine of angle between respective vectors

applications:

- train on 2K pages of English text, achieved average score on synonym portion of TOEFL
- train on introductory psychology textbook, achieved passing score on multiple-choice exam

Plates for Graphical Models

Probabilistic representations of documents: start with plate notation

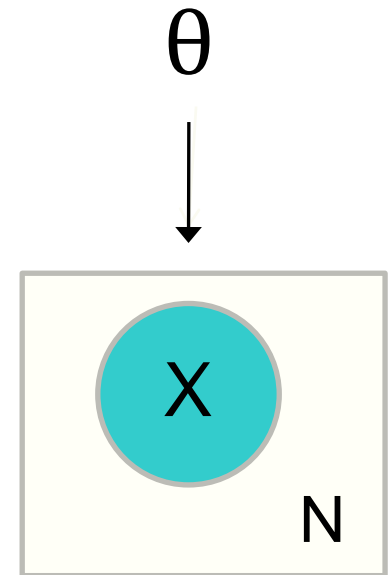
Example: coin with unknown bias

θ = probability of heads (parameter)

X = coin toss outcome

N observations (repetitions)

$$P(X = H \mid \theta) = \theta$$



Observe: T T H T T T H T

ML: $\theta = 1/4$

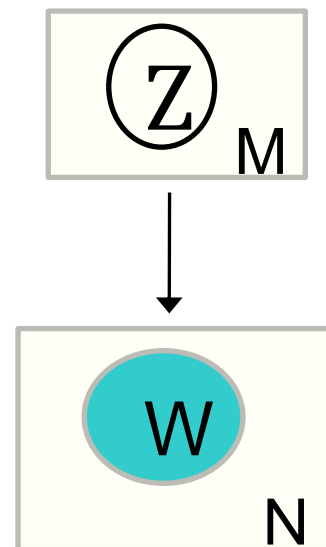
Simple Probabilistic Topic Models

Unigram model - each word with its own probability of appearing in document of length N



Problem: does not represent document containing a set of topics

Mixture of unigrams



Probabilistic LSI

Topic (aspect) model [Hoffman, 99]: probabilistic model of word production

$$P(w, d) = \sum_k P(d | z_k) P(w | z_k) P(z_k)$$

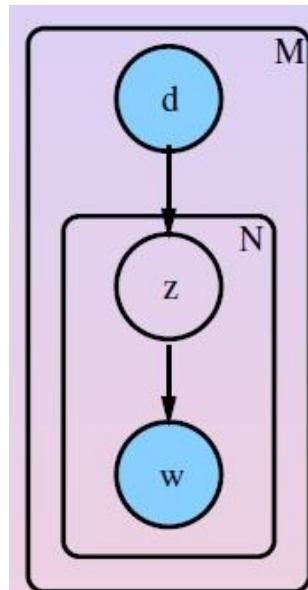
Generative model:

- select document d with probability $P(d)$
- select latent topic z with probability $P(z|d)$: $\text{Mult}(z_k | \theta_k^{d_i})$
- generate word with probability $P(w|z)$:

$$\text{Mult}(w_j | \phi_j^k)$$

Problems

- Lots of parameters - mixture parameters for each document
- Does not generalize well



Conjugate Distributions

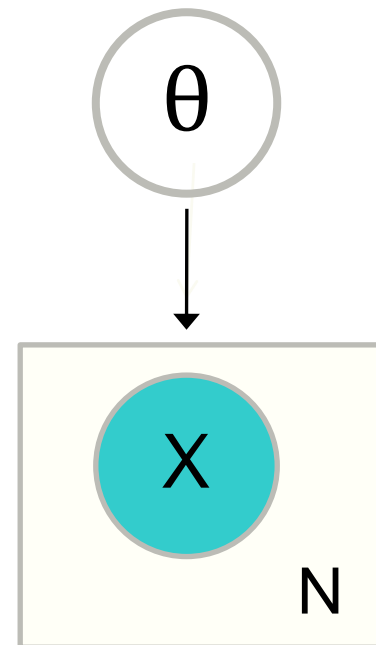
To improve generative model, need to understand conjugate distributions

X = coin toss outcome (Bernoulli)

N observations (repetitions)

Prob of observing n heads:

$$P(n | \theta, N) \propto \theta^n (1 - \theta)^{N-n}$$



Prior over θ : Beta(α, β) [think of α as count of heads; β as count of tails]

θ = probability of heads (variable)

Key property: posterior is same form as prior

Conjugate Distributions

Prior: pseudo-observations of heads/tails:

$$\text{Beta}(\theta | \alpha, \beta) = \frac{\theta^{\alpha-1} (1-\theta)^{\beta-1}}{B(\alpha, \beta)}$$

After n heads and $N-n$ tails, posterior another Beta distribution, with a change in parameters:

$$P(\theta | n, N, \alpha, \beta) = \frac{P(n, N | \theta)P(\theta | \alpha, \beta)}{\int P(n, N | \theta')P(\theta' | \alpha, \beta)d\theta'}$$

$$\propto [\theta^n (1-\theta)^{N-n}] [\theta^{\alpha-1} (1-\theta)^{\beta-1}]$$

$$\propto \theta^{n+\alpha-1} (1-\theta)^{N-n+\beta-1}$$

Conjugate Distributions

Prior $P(\theta)$ is conjugate to class of likelihood if resulting posterior is in same family as $P(\theta)$

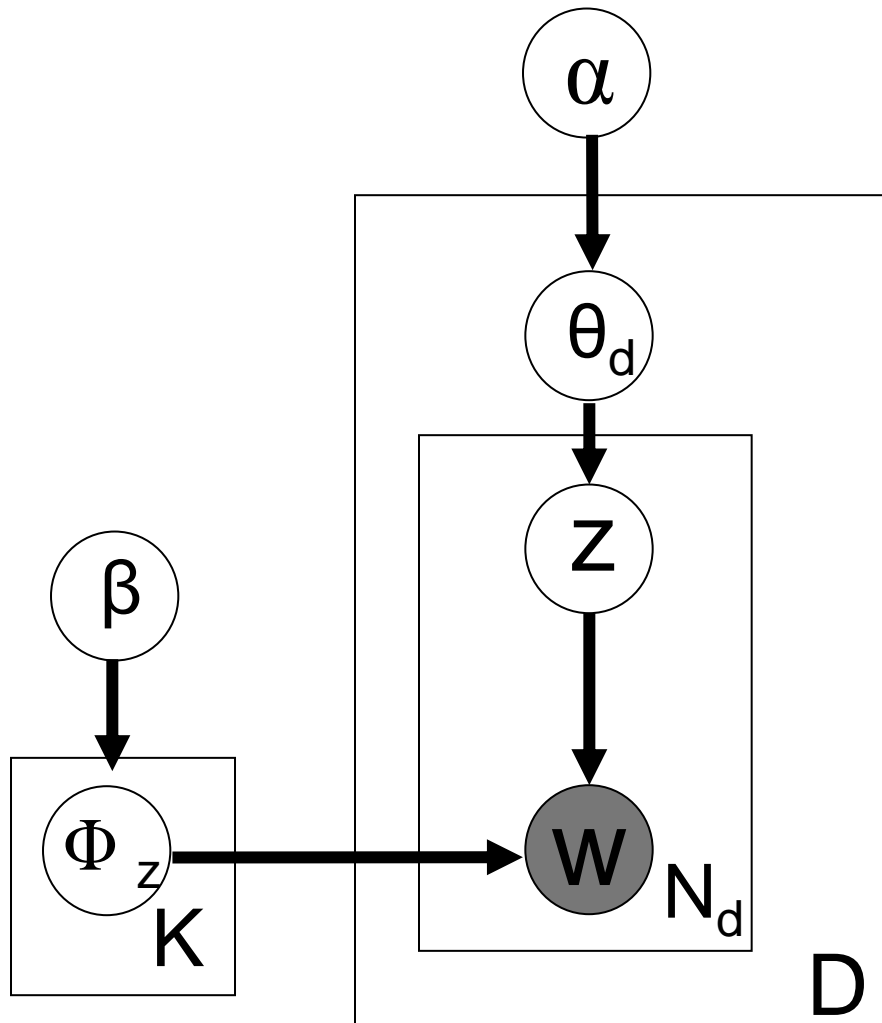
$$P(\theta | X) = \frac{P(X | \theta)P(\theta)}{\int P(X | \theta')P(\theta')d\theta'}$$

Important because it avoids integration required to calculate posterior

Other conjugate distributions (all exponential family distributions have conjugate priors), e.g., [Likelihood-Prior-Posterior]: Gaussian-Gaussian-Gaussian; Poisson-Gamma-Gamma; Multinomial-Dirichlet-Dirichlet

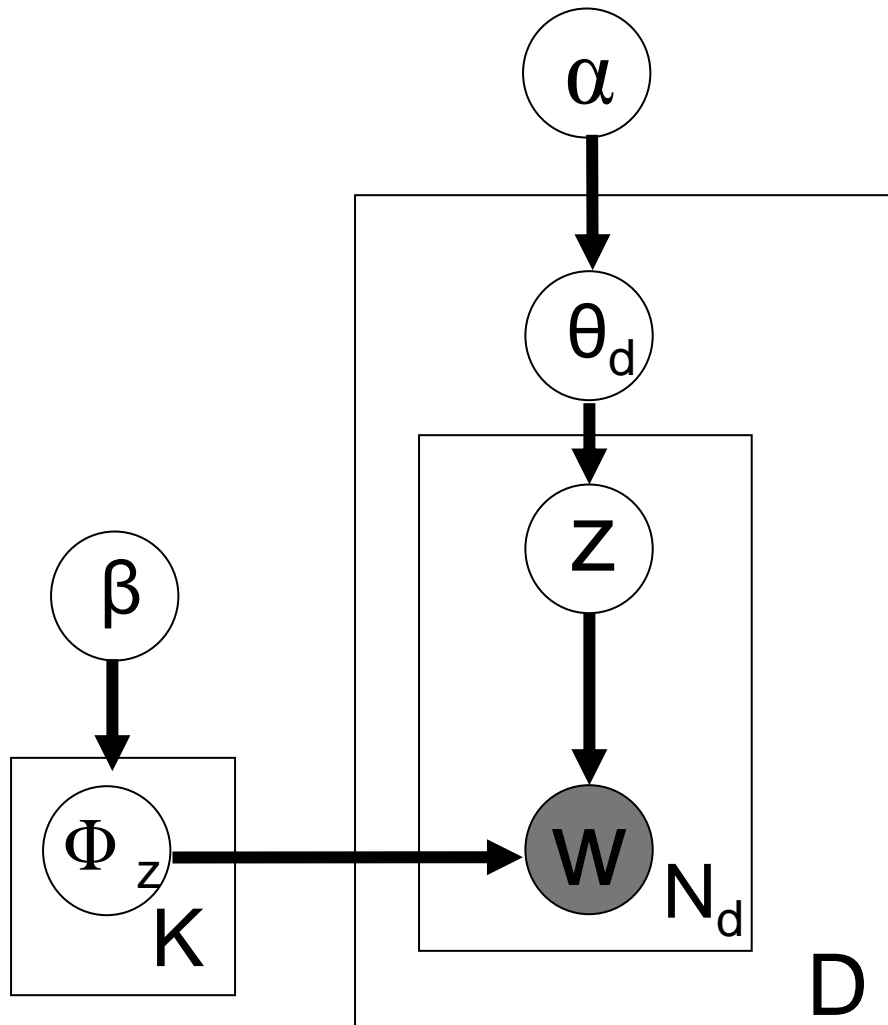
(Dirichlet generalizes Beta to K alternatives)

Latent Dirichlet Allocation



- α – Dirichlet prior on θ_d
- β – Dirichlet prior on Φ_z
- D – number of documents
- N_d – Number of words in document d .
- K – number of latent topics.
- θ_d – distribution of topics in document d .
- z – latent topic
- w – observed word
- Φ_z – distribution of words generated from topic z .

Latent Dirichlet Allocation



Generative process:

Choose $\theta_d \sim \text{Dir}(\alpha)$

For each of N_d words w :

Choose topic $z \sim \text{Mult}(\theta_d)$

Choose word $w \sim \text{Mult}(\Phi_z)$

Cleaner generative model with fewer parameters than PLSI/aspect model

Inference in LDA

- Tricky to compute posterior over hidden variables given a document:

$$P(z | w) = \frac{P(z, w)}{P(w)}$$

- Numerator is tractable, due to conjugacy:

$$P(z, w) = P(w | z)P(z) = P(w | z) \int P(z | \theta)P(\theta | \alpha) d\theta$$

since $P(z|\theta)$ is Multinomial, $P(\theta|\alpha)$ is Dirichlet, product is Dirichlet, integral is expected value of Dirichlet

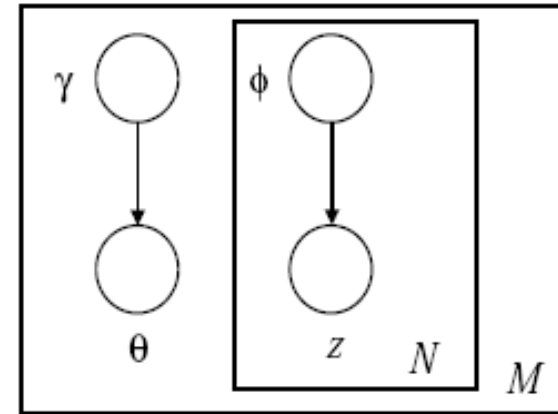
- but denominator not tractable, since it involves summing over all z
- note that document represented as continuous mixture, with

$$P(w | \alpha, \beta) = \int P(\theta | \alpha) \left(\prod_{n=1}^N P(w_n | \theta, \beta) \right) d\theta$$

Variational Approximation for LDA

- γ correspond to parameters of Dirichlet distribution over θ given observed words
- Φ corresponds to probability that n th word generated by topic i

$$q(\theta, z | \gamma, \phi) = q(\theta | \gamma) \prod_{n=1}^N q(z_n | \phi_n)$$



- Use variational EM: minimize KL between true posterior $p()$ and variational distribution, updates:

$$\gamma_i = \alpha_i + \sum_n \phi_{ni}$$

$$\phi_{ni} \propto \beta_{i w_n} \exp\{E_q[\log(\theta_i) | \gamma_i]\}$$

Variational Approximation for LDA

- Must normalize the multinomial: $\sum_i \phi_{ni} = 1$
- Updates make sense
 - γ : posterior Dirichlet based on expected observations under variational distribution $E[z|]$
 - Φ : multinomial update based on Bayes theorem:

$$p(z_n | w_n) \propto p(w_n | z_n) p(z_n)$$

- Note that each update depends on other set of variational parameters, so need to alternate between these two updates until bound converges

Collapsed Gibbs Sampling for LDA

- The latent topics, z , are sampled
- The distributions θ and Φ are integrated out
- Closed form sampling equations

$$\Pr(z_{d,i} = z \mid \mathbf{z}_{-(d,i)}, \mathbf{W}) \propto (N_{d,z}^{DT} + \alpha) \frac{(N_{z,w}^{TW} + \beta)}{\sum_{w'} (N_{z,w'}^{TW} + \beta)}$$

- Each iteration requires $O(K \cdot \text{corpus size})$ ops.
- **EXPENSIVE** when counts are large
- Overall - slower, more accurate than variational inference