# Chapter 5 Advanced Text Representations

# Overview

- Latent semantic topic models: **probabilistic models of content**
  - Probabilistic Latent Semantic Analysis (pLSA)
  - Latent Dirichlet Allocation (LDA)
- Approximate inference: examples of use of:
  - Expectation maximization algorithm
  - Variational inference
  - Gibbs sampling
- Integration of the probabilistic topic models in **retrieval models**

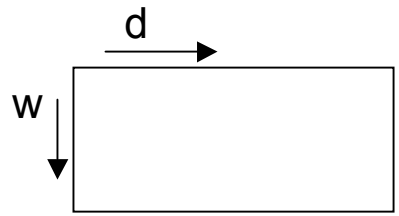
# Latent semantic topic models

= a class of unsupervised (or semi-supervised) models in which the semantic properties of words and documents are expressed in terms of topics

models are also called **aspect models**

Latent Semantic Indexing:

the semantic information can be derived from a word-document matrix




But, LSI is unable to capture multiple senses of a word

**Probabilistic topic models**

# Panini

Panini = Indian grammarian (6th-4thcentury B.C. ?) who wrote a grammar for **sanskrit**

**Realizational chain** when creating natural language texts:

Ideas -> broad conceptual components of a text -> subideas -> sentences -> set of semantic roles-> set of grammatical and lexical concepts ->character sequences

# Probabilistic topic model

= **Generative model** for documents: probabilistic model by which documents can be generated

document = probability distribution over topics

topic = probability distribution over words

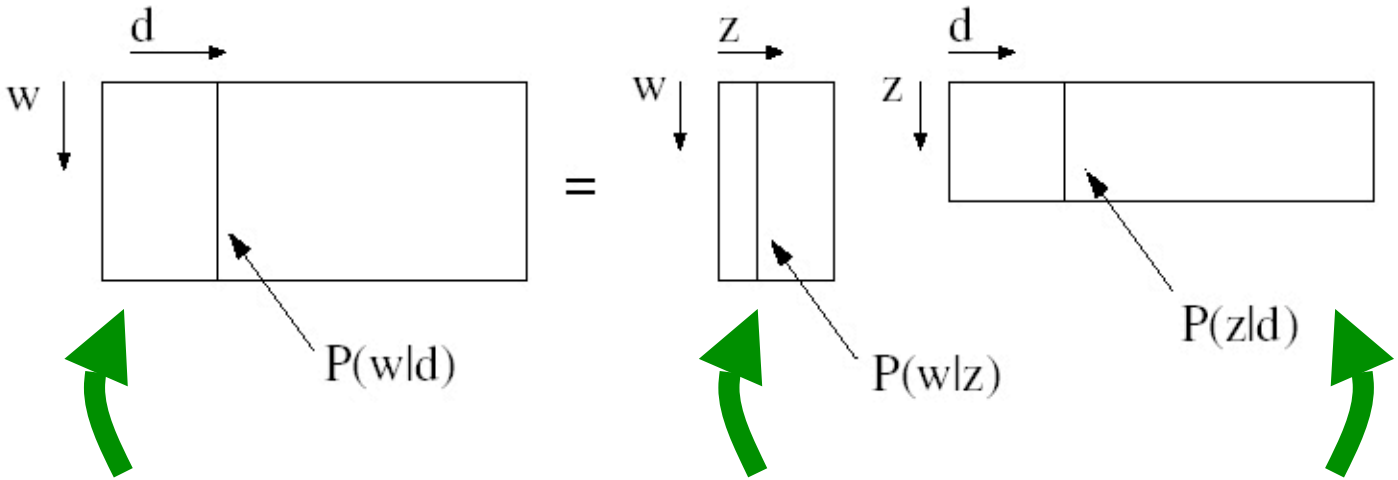
To make a new document, one chooses a distribution over topics, for each topic one draws words according to a certain distribution:

select a document $d_j$ with probability $P(d_j)$

pick a **latent class $z_k$** with probability $P(z_k|d_j)$

generate a word $w_i$ with probability $P(w_i|z_k)$

[Steyers & Griffiths 2007]




observed word distributions

word distributions per topic

topic distributions per document

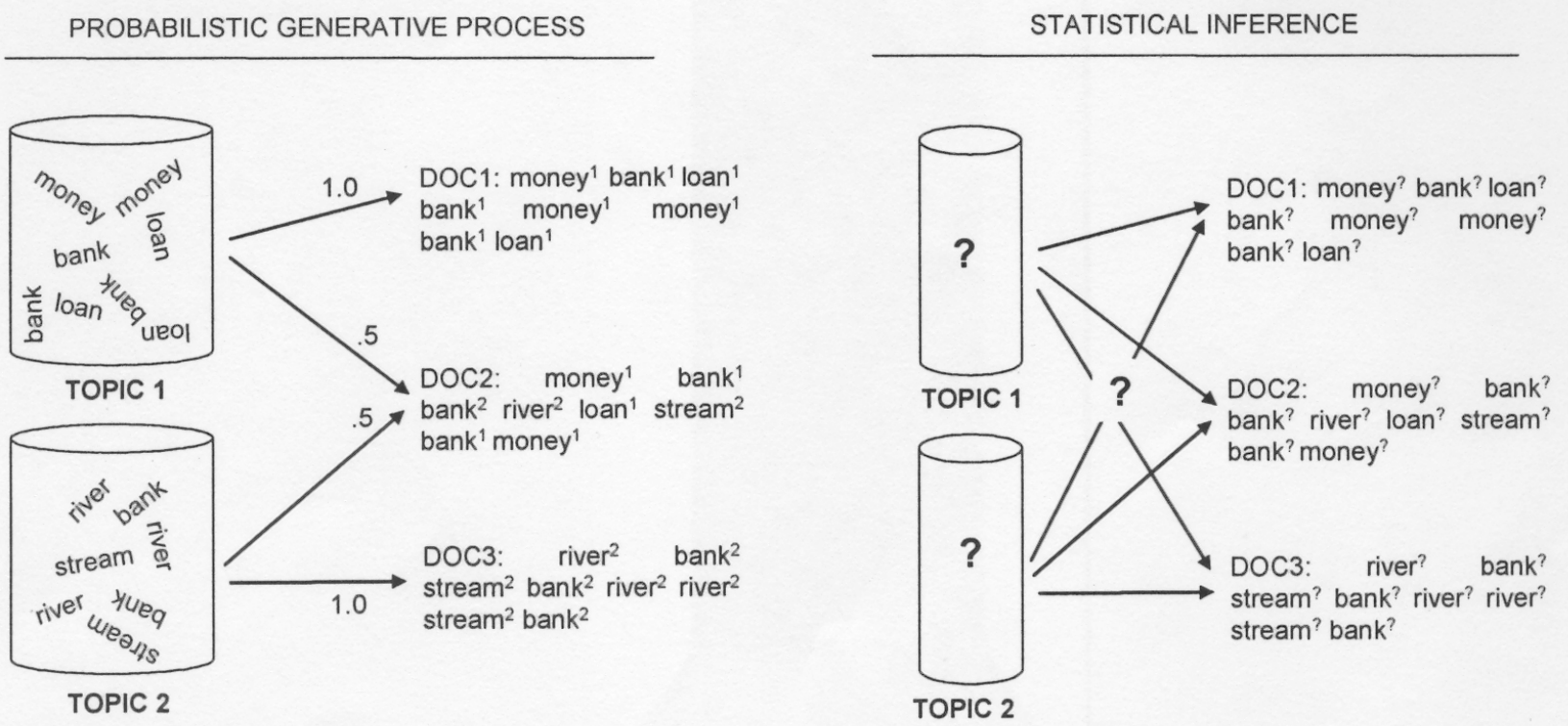
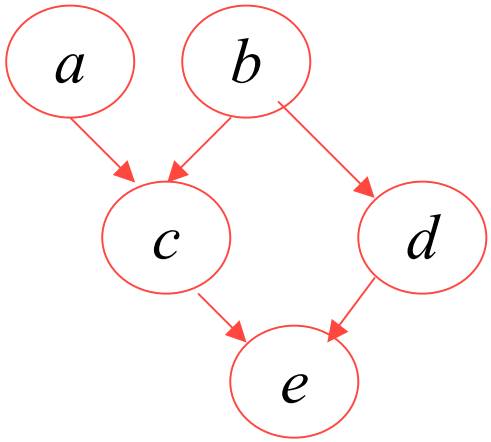



**Figure 2.** Illustration of the generative process and the problem of statistical inference underlying topic models

[Steyvers & Griffiths 2007]

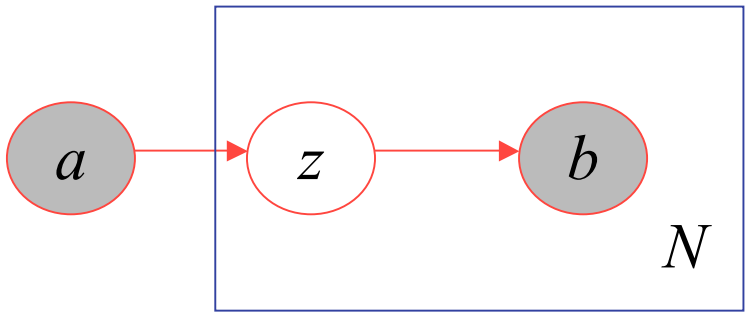
# Probabilistic graphical models

- Node represents a random variable (or group of random variables)
- Edge represents probabilistic relationships between these variables
- **Bayesian network:**
  - directed graphical model



# Plate notation

- Plate:
  - graphical notation that allows multiple nodes to be expressed more compactly
  - represents *N* nodes of which only a single example is shown




- shaded node: observed variable
- non-shaded node: hidden variable
- directed edge: conditional dependency of head node on tail node

- Observed node/variable:
  - we know the current value, as it is observable

- Hidden node/variable:
  - variable whose state can not be observed: its existence and value **can only be inferred** by observing the outcome of the observed variables

- Imagine a graphical model, describing a process with observable nodes $X$ and hidden nodes $\Theta$ where the dependency probabilities of $X$ on $\Theta$ are unknown
- The process runs for several times, which yields a collection of observed variables $x$: this can be used to estimate the values $\theta$ of variables $\Theta$
- The best guess that can be made about $\theta$, is to claim that since it generated the observations $x$, this $x$ is the most likely outcome of $\theta$
- In terms of probabilities: we are looking for the value of $\theta$ that gives the highest probability for $P(x \mid \theta)$: called $L(\theta \mid x)$ or the **likelihood** of $\theta$ given $x$

- **Maximum Likelihood Estimation** (MLE):

$$\theta^* = \arg\max_{\theta} P(x|\theta)$$

- If the model is relatively simple, the maximum can be searched analytically, using the formula:

$$\frac{dP(\theta|x)}{d\theta} = 0$$

- Problems in complex models:
  - Most likely values of parameters: maximum likelihood of model
  - Exact likelihood with dependence of variables: likelihood impossible to calculate in full

- We have to **approximate** the calculations: e.g., by
  - **Expectation Maximization** algorithm: iterative method to estimate the probability of unobserved, latent variables: until local optimum is obtained
  - **Variational inference**: approximate model by easier one
  - **Gibbs sampling**: update parameters sample-wise

See below examples of these approximate inference methods

- If $\theta$ has a known prior distribution (i.e. $P(\theta)$ is known): **Maximum A Posteriori Estimation** (MAP)

$$\theta^* = \arg\max_{\theta} P(x|\theta)P(\theta)$$

- If a conditional dependency is said to follow a specific distribution, a **conjugate prior** is placed over the parameters: If $P(\theta)$ is a conjugate prior to $P(x|\theta)$,

it makes $\frac{dP(\theta|x)P(\theta)}{d\theta} = 0$ much easier to calculate

# Probabilistic Latent Semantic Analysis (pLSA)

## Generative story for pLSA

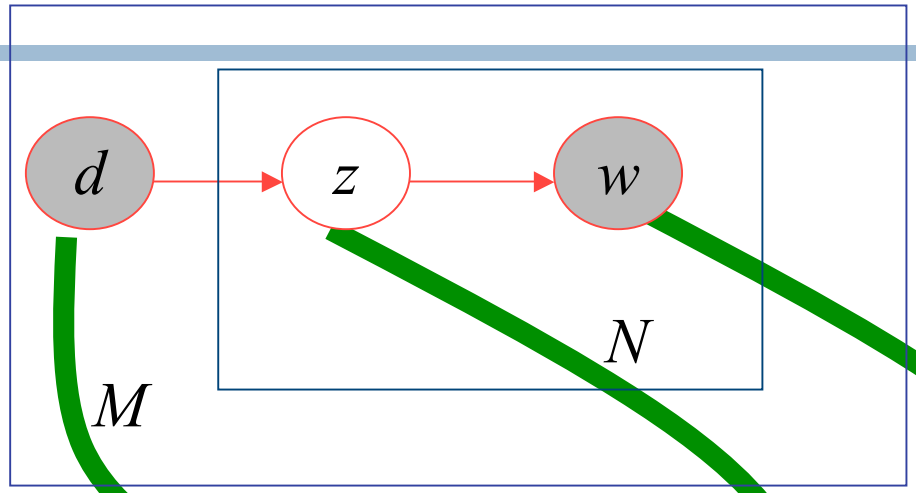
For each $j$ of the $M$ documents:

Select a document $d_j$ with probability $P(d_j)$

For each word position $i$ of the $N$ word positions of $d_j$:

Choose a **latent class $z_k$** with probability $P(z_k|d_j)$
Choose a word $w_i$ with probability $P(w_i|z_k)$

# Probabilistic Latent Semantic Analysis (pLSA)




[Hofmann 1999]

*M* = number of documents
*N* = number of words
(in terms of word positions)

John goes into the building, sits down waitress shows him menu. John orders. The waitress brings the food. John eats quickly, puts $10 on the table and leaves. ..

John goes the park with the magnolia trees and meets his friend, ...

Topic 1

waitress
$ food
Menu
...

Topic 2

park
Tree
...

....

# Probabilistic Latent Semantic Analysis

Translating the document or text generation process into a joint probability model:

$$P(d_j, w_i) = P(d_j)P(w_i|d_j)$$

$$P(w_i|d_j) = \sum_{k=1}^{K} P(w_i|z_k)P(z_k|d_j)$$

where $K$ = number of topics (**a priori defined**)

**Training:** maximizing the likelihood function:

$$L = \prod_{j=1}^{M}\prod_{i=1}^{N} P(d_j, w_i)^{n(d_j, w_i)} = \sum_{j=1}^{M}\sum_{i=1}^{N} n(d_j, w_i)\log P(d_j, w_i)$$

where $n(d_j, w_i)$ = frequency of $w_i$ in $d_j$
(e.g. trained with EM algorithm)

# Probabilistic Latent Semantic Analysis

Initial estimates of the parameters $P(w_i|z_k)$ and $P(z_k|d_j)$

**E-step**: posterior probabilities are computed for the latent variables *z*, based on the current estimates of the parameters:

$$P(z_k|d_j,w_i) = \frac{P(w_i|z_k)P(z_k|d_j)}{\sum_{l=1}^{K} P(w_i|z_l)P(z_l|d_j)}$$

**M-step**: parameters are re-estimated in order to maximize the likelihood function:

$$P(w_i|z_k) = \frac{\sum_{j=1}^{M} n(d_j,w_i)P(z_k|d_j,w_i)}{\sum_{l=1}^{N}\sum_{j=1}^{M} n(d_j,w_l)P(z_k|d_j,w_l)}$$

$$P(z_k|d_j) = \frac{\sum_{i=1}^{N} n(d_j,w_i)P(z_k|d_j,w_i)}{\sum_{l=1}^{K}\sum_{i=1}^{N} n(d_j,w_i)P(z_l|d_j,w_i)}$$

Iterating the E-step and M-step defines a convergent procedure that approaches a local maximum

- **Disadvantages of the pLSA model**:
  - learns $P(z_k|d_j)$ only for those documents on which it is trained except for some limited folding in
    - folding in: repeat EM by clamping the word-topic model
  - number of hidden variables (topics) to learn grows linearly with the growth of the number of documents

# Latent Dirichlet Allocation

## Generative story for LDA

For each $d_j$ of the $M$ documents

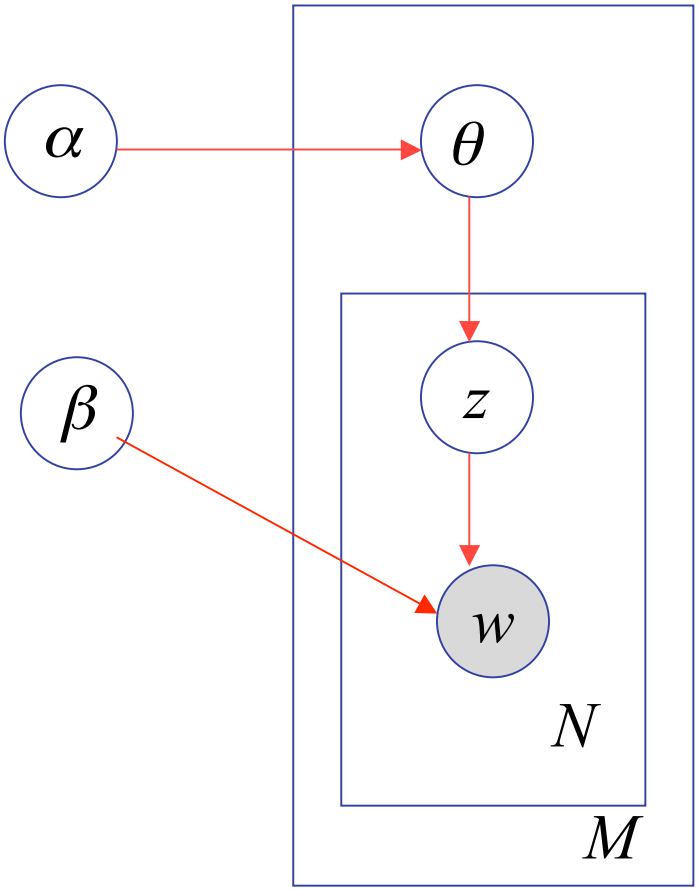
Choose $\theta \sim Dir\,(\alpha)$

For each word position $i$ of the $N$ word positions of $d_j$:

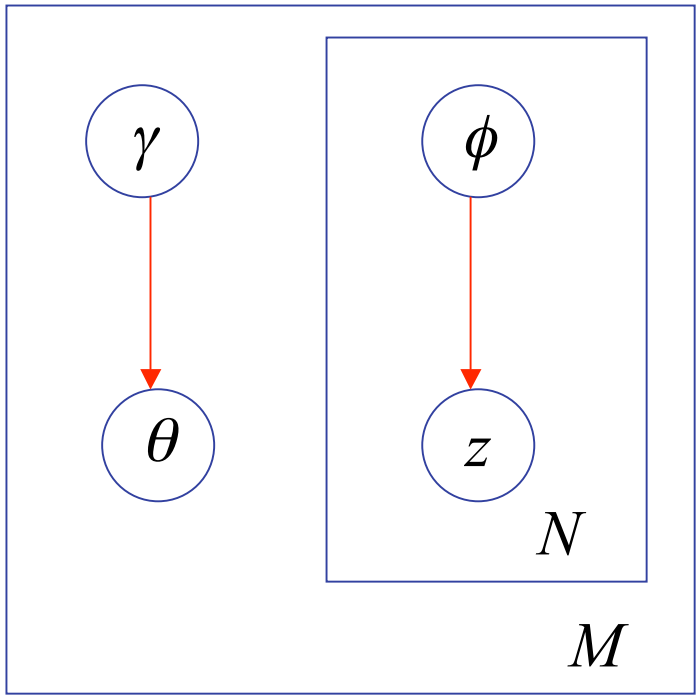
Choose a **latent topic** $z_k \sim Multinomial\,(\theta)$

Choose a word $w_i$ from $P(w_i | z_k, \beta)$, a multinomial probability distribution conditioned on topic $z_k$

# Latent Dirichlet Allocation




(1)




(2)

[Blei et al. JMLR 2003]

# Latent Dirichlet Allocation

- **Latent Dirichlet Allocation** (LDA) treats topic mixture weights as a *k*-parameter hidden random variable $\theta$
- **Training**
  - Key inferential problem: computing the distribution of the hidden variables $\theta$ and $z$ given a document , i.e., $P(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta)$: intractable for exact inference (model1)
  - $\alpha$: Dirichlet prior, can be interpreted as a prior observation count for the number of times a topic is sampled in a document, before having observed any actual words from that document

# Latent Dirichlet Allocation

- Model 2 = simple modification of the original graphical model 1: the chain $\alpha \rightarrow \theta \rightarrow z$ is replaced by $\gamma \rightarrow \theta$ and $\phi \rightarrow z$
- Compute approximation of model 1 by model 2 for which the KL divergence between the two models $p(\theta,z|\gamma,\phi)$ and $P(\theta,z|\ w,\alpha,\beta)$ is minimal
- Iterative updating of $\gamma$ (topic distribution) and $\phi$ (word distribution) for each document and recalculation of corpus-level variables $\alpha$ and $\beta$ by means of EM algorithm

- Inference for new document:
  - Given $\alpha$ and $\beta$: we infer $\gamma$ (topic distribution) and $\phi$ (word distribution) of the new document

# Dirichlet prior

- Prior:
  - captures our initial uncertainty about the parameters
  - e.g., conjugate prior
- One such prior is the **Dirichlet distribution**, which which is characterized by a set of hyperparameters $\alpha_1,..., \alpha_k$, so that:
  - $\theta \sim$ Dirichlet($\alpha_1,..., \alpha_k$) if $P(\theta) \propto \prod_{k=1}^{K} \theta_k^{\alpha_k - 1}$

- Variational inference requires a simplified model to approximate the real one: calculation is based on variables whose dependencies have been removed: altering the original model

- **Gibbs sampling** is an inference technique that works directly with the original model
  - = simple widely applicable **Markov Chain Monte Carlo** (MCMC) algorithm
  - A Markov chain is a random process, consisting of different states
  - Each state *j* has a probability $P_{ij}$ of being reached from state *i* ($P_{ij}$ can be 0), called the transition probability: a Markov chain has the first order Markov property in that $P_{ij}$ is only dependent on state *i*

# Gibbs sampling

- Several Monte Carlo Markov Chain algorithms are possible that perform a random walk over a Markov chain: e.g., **Gibbs sampling**
  - The states that are reachable from a given state are those where only one variable differs in value
  - The values of other variables are held fixed, and the transition probabilities are the posterior probabilities for the updated variable
  - By cycling through each variable until convergence, the **equilibrium state** is reached.
  - The samples are then coming from the full joint distribution

# Gibbs sampling

- $P(\boldsymbol{z}) = P(z_1,...,z_L)$: distribution from which we want to sample
- Consider initial state for the Markov chain
- Each step of the Gibbs sampling: replaces the value of one of the variables by a value drawn from the distribution of that variable conditioned on the values of the remaining variables
- Procedure is repeated either by cycling through the variables in a particular order, or by choosing the variable to be updated at each step at random from the distribution
- After burn-in period, samples are saved at regular intervals

# Gibbs sampling

1. Initialize $\{z_i : i = 1,...,L\}$
2. For $\tau = 1,...,T$
   - Sample $z_1^{(\tau+1)} \sim P(z_1 | z_2^{(\tau)}, z_3^{(\tau)},...,z_L^{(\tau)})$
   - Sample $z_2^{(\tau+1)} \sim P(z_2 | z_1^{(\tau+1)}, z_3^{(\tau)},...,z_L^{(\tau)})$

   .
   .
   .

   - Sample $z_j^{(\tau+1)} \sim P(z_j | z_1^{(\tau+1)},..., z_{j-1}^{(\tau+1)}, z_{j+1}^{(\tau)},...,z_L^{(\tau)})$

   .
   .
   .

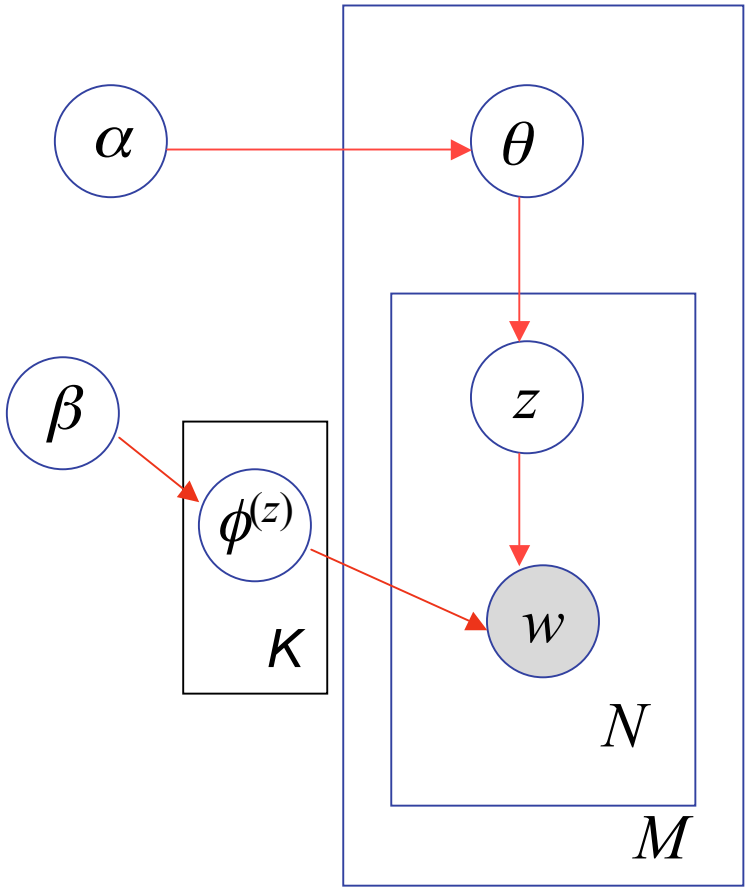
   - Sample $z_L^{(\tau+1)} \sim P(z_L | z_1^{(\tau+1)}, z_2^{(\tau+1)},...,z_{L-1}^{(\tau+1)})$

- The sampling is done sequentially and proceeds until the sampled values approximate the target distribution

# Gibbs sampling

- Estimating the LDA model is here done with Gibbs sampling
- The Gibbs sampling procedure
  - considers each word token in the text collection in turn
  - estimates the probability of assigning the current word token to each topic, conditioned on the topic assignments of all other word tokens
  - from this conditional distribution, a topic is sampled and stored as the new topic assignment for this word token

# Latent Dirichlet Allocation




[Steyvers & Griffiths 2007]

# Latent Dirichlet Allocation

- Use of a symmetric Dirichlet prior for $\alpha$ and $\beta$ (smoothed topic distribution):
  - Symmetric Dirichlet distribution, where all of the elements making up the vector $\alpha$ or $\beta$ have the same value)
  - Good estimates depend on the number of topics ($K$) and the vocabulary size ($N$): e.g., $\alpha = 50/K$ and $\beta = 200/N$ (often $\beta = 0.01$ in a realistic text collection)

# Gibbs sampling

- $C^{NK}_{NxK}$ and $C^{MK}_{MxK}$: matrices of counts
- Gibbs sampling considers each word token $w_i$ in the text collection in turn
- $C^{NK}_{wik}$ = number of times some word token $w$ is assigned to topic $k$, not including the current word $i$
- $C^{MK}_{dik}$ = number of times topic $k$ is assigned to some word token $w$ in $d$, not including the current word $i$
- We compute:

$$P(z_i = k | \mathbf{z}_{-i}, w_i, d_i, .) \propto \frac{C^{NK}_{wik} + \beta}{\sum_{w=1}^{N} C^{NK}_{wk} + N\beta} \frac{C^{MK}_{dik} + \alpha}{\sum_{l=1}^{K} C^{MK}_{dil} + K\alpha}$$

where

$z_i = k$ represents the topic assignment of word $i$ to topic $k$

$\mathbf{z}_{-i}$ = topic assignment of all other word tokens

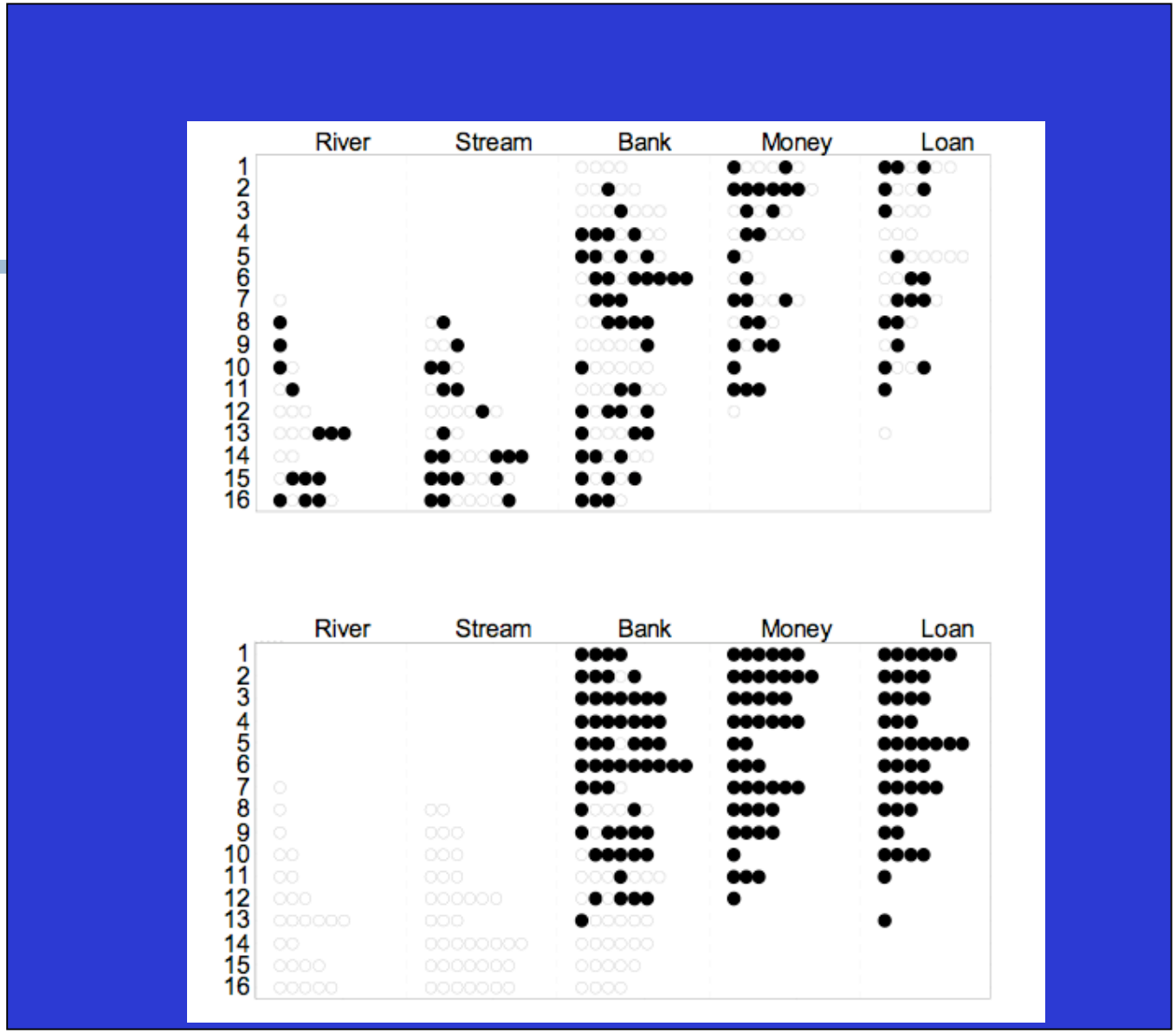
"." refers to all other observed or known information, such as all other words and document indices $\boldsymbol{w}_{-i}$ *and* $\boldsymbol{d}_{-i}$ and the hyperparameters $\alpha$ and $\beta$

- Normalize by the sum over all topics

- Count matrices are updated
- Sampling and updating are iterated until the equilibrium state is reached
- The sampling algorithm gives direct estimates of *z* for every word token *i*
- Many applications require estimates of $\phi'$ (word-topic distributions) and $\theta'$ (topic-document distributions), which can be obtained from the above count matrices:

$$\phi'^{(k)}_i = \frac{C^{NK}_{ik} + \beta}{\sum_{l=1}^{N} C^{NK}_{lk} + N\beta}$$

$$\theta'^{(d)}_k = \frac{C^{MK}_{dk} + \alpha}{\sum_{l=1}^{K} C^{MK}_{dl} + K\alpha}$$




**An example of the results of applying the LDA model trained with a Gibbs sampling procedure.**



[Steyvers & Griffiths 2007]

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 |

**Figure 1.** An illustration of four (out of 300) topics extracted from the TASA corpus.

[Steyvers & Griffiths 2007]

# Determining the number of topics

- The above models rely on a number of topics a priori set:
  - Models that dynamically allocates topics:
    - **Chinese Restaurant Franchise** topic model, which uses a stochastic mechanism called the Chinese Restaurant Process

[Blei, Griffiths, Jordan & Tenenbaum, NIPS 2003]

# Probabilistic topic models

- **Probabilistic models of text generation** (cf. model of text generation by Panini)
- **Understanding** by the machine = we **infer the latent structure** from which the document/text is generated

- Today:
  - Based on bag-of-words representations
  - Addition of other structural information is currently limited (e.g., syntax information)
  - But, acknowledged **potential for richly structured statistical models of language and text understanding** in general

# Probabilistic topic models in retrieval models

- Consider incorporation of a probabilistic topic model into a:
  - A language retrieval model
  - An inference network retrieval model

# Language retrieval model

- Integration of pLSA or LDA in language retrieval model: e.g.,

$$P(q_1,\ldots,q_m|D) = \prod_{i=1}^{m}(\lambda P(q_i|D) + (1-\lambda)P(q_i|C))$$

where

$$P(q_i|D) = \sum_{k=1}^{K} P(q_i|z_k)P(z_k|D)$$

computed with latent topic model
and $K$ = number of topics

# What have we learned?

- Advanced text representations offer possibility to probabilistically model content and to integrate the models into retrieval models:
  - **language models and inference net models**
- Important:
  - **concept of latent topics**
  - **approximate inference methods for parameter estimations in Bayesian networks: EM and Gibbs sampling**

# Research questions to be solved

- Further investigations into probabilistic content models of retrievable objects:
  - More complex hierarchical structures
  - Integration of other latent variables
  - Integration of good initialization methods for approximate inference algorithms especially when a large number of variables has to be estimated
  - Integration of limited supervision and other external knowledge

## Further reading


Blei, D.M., Ng, A.Y. and Jordan, M.I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3, 993-1022.

Blei, D. M., Griffiths, T. L., Jordan, M. I., and Tenenbaum, J. B. (2003). Hierarchical topic models and the nested Chinese restaurant process. In *Proceedings NIPS*.

Hofmann, T. (1999). Probabilistic latent semantic indexing. In *Proceedings of Twenty-second Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval.*

Koller, D. & Friedman, N. (2010). *Probabilistic Graphical Models: Principles and Techniques.* The MIT Press.

Steyvers, M. and Griffiths, T. (2007). Probabilistic topic models. In T. Landauer, D.McNamara, S. Dennis& W. Kintsch (Eds.), *The Handbook of Latent Semantic Analysis*. Lawrence Erlbaum Associates.