Statistical Learning Models for Text and Graph Data
Topic Models

Yangqiu Song

Hong Kong University of Science and Technology

yqsong@cse.ust.hk

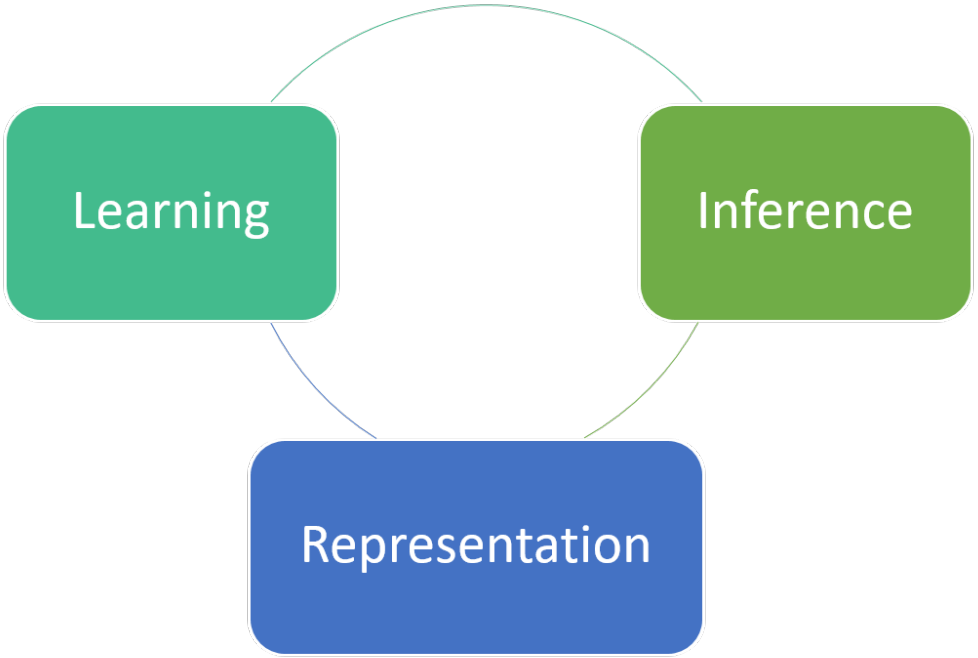
October 23, 2019

∗Contents are based on materials created by Noah Smith, Xiaojin (Jerry) Zhu, Chengxiang Zhai

Reference Content

- Noah Smith. CSE 517: Natural Language Processing `https://courses.cs.washington.edu/courses/cse517/16wi/`
- Xiaojin (Jerry) Zhu. CS 769: Advanced Natural Language Processing. `http://pages.cs.wisc.edu/~jerryzhu/cs769.html`
- Chengxiang Zhai. CS598CXZ Advanced Topics in Information Retrieval. `http://times.cs.uiuc.edu/course/598f16/`

Course Organization



- Representation: language models, word embeddings, topic models, knowledge graphs
- Learning: supervised learning,unsupervised learning, semi-supervised learning, distant supervision, indirect supervision, sequence models, deep learning, optimization techniques
- Inference: constraint modeling, joint inference, search algorithms

Overview

Paragraphs of Text

- A language model is a probability distribution over $\mathcal{V}^\dagger$
- Typically P decomposes into probabilities $P(x_i|\mathbf{h}_i)$
  - We considered n-gram, log-linear, and neural language models, etc.

Three Kinds of Cotext

If we consider a word token at a particular position i in text to be the observed value of a random variable X_i, what other random variables are predictive of/related to X_i?

- The words that occur within a small "window" around i (e.g., $x_{i-2}, x_{i-1}, x_{i+1}, x_{i+2}$, or maybe the sentence containing i) $\rightarrow$ distributional semantics
- The document containing i (a moderate-to-large collection of other words) $\rightarrow$ topic models
- A sentence known to be a translation of the one containing i $\rightarrow$ translation models

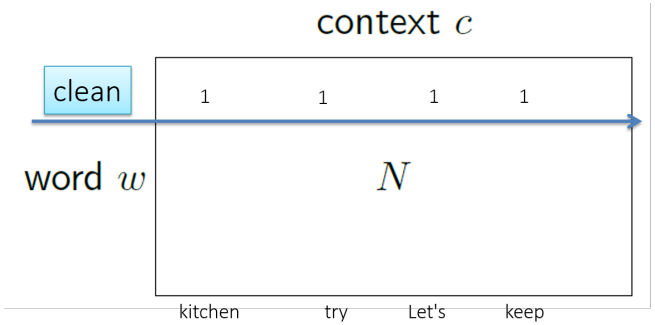
Overview

Topic Models

- Words are not independent and identically distributed (i.i.d.)!
  - Predictable given history: n-gram/Markov models
  - Predictable given other words in the document: topic models
- Let $Z = \{1, \ldots, k\}$ be a set of "topics" or "themes" that will help us capture the interdependence of words in a document
  - Usually these are not named or characterized in advance; they are just k different values with no a priori meaning

The Term-Document Matrix

- Let $\mathbf{A} \in \mathbb{R}^{V \times M}$ contain statistics of association between words in $\mathcal{V}$ and M documents. N is the total number of word tokens.
- Comparison of contexts
  - Local context (Let's try to keep the kitchen clean.)



  - Document-level context ($[\mathbf{A}]_{v,d} = c_{\mathbf{x}_d}(v)$)
    - d1: "yes, we have no bananas"
    - d2: "say yes for bananas"
    - d3: "no bananas , we say"

	1	2	3
,	1	0	1
bananas	1	1	1
for	0	1	0
have	1	0	0
no	1	0	1
say	0	1	1
we	1	0	1
yes	1	1	0

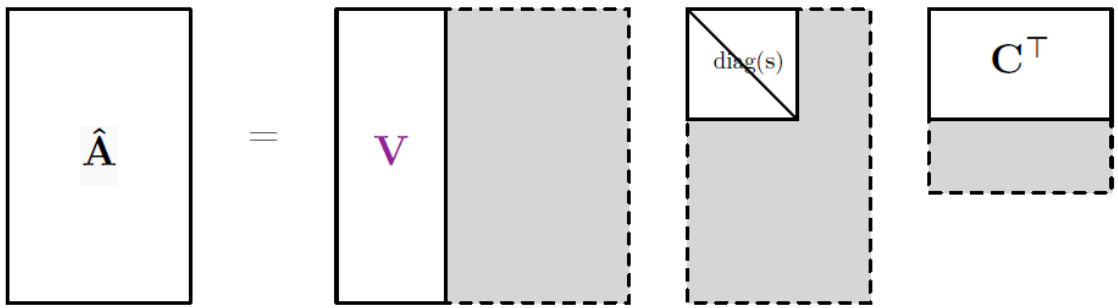
Topic Models: Latent Semantic Indexing/Analysis (Deerwester et al. (1990))

- LSI/A seeks to solve:

$$\underset{V \times M}{\mathbf{A}} \approx \underset{V \times d}{\mathbf{V}} \times \underset{d \times d}{\text{diag}(\mathbf{s})} \times \underset{d \times M}{\mathbf{C}}^{\top}$$

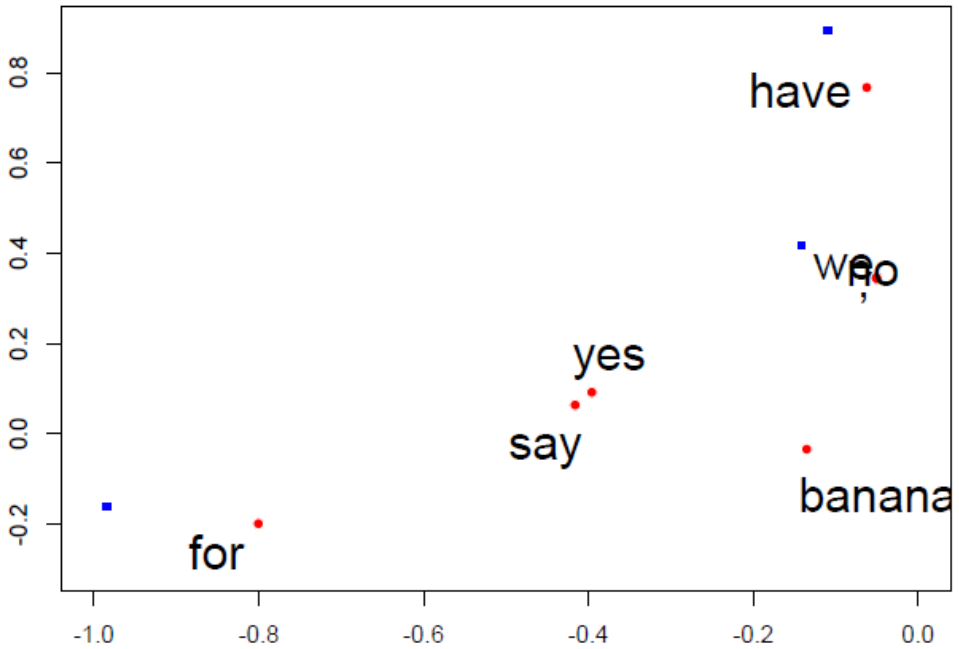
  where $\mathbf{V}$ contains embeddings of words and $\mathbf{C}$ contains embeddings of documents

- This can be solved by applying singular value decomposition to $\mathbf{A}$



LSI/A Example

- $d = 2$: Words and documents in two dimensions.



	1	2	3
,	1	0	1
bananas	1	1	1
for	0	1	0
have	1	0	0
no	1	0	1
say	0	1	1
we	1	0	1
yes	1	1	0

Note how "no", "we", and "," are all in the exact same spot. Why?

Understanding LSI/A

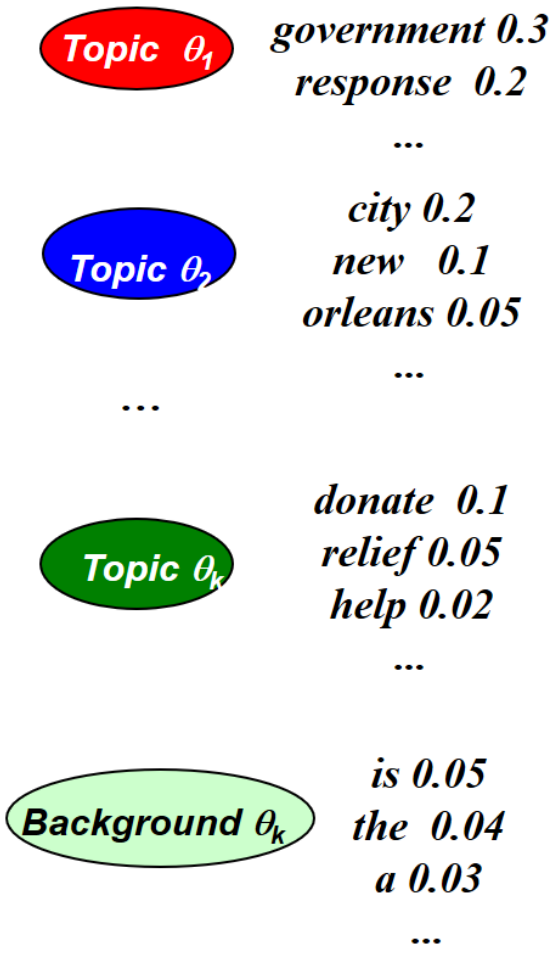
- Mapping words and documents into the same d-dimensional space.
- Bag of words assumption (Salton et al. (1975)): a document is nothing more than the distribution of words it contains.
- Distributional hypothesis (Harris (1954); Firth (1957)): words are nothing more than the distribution of contexts (here, documents) they occur in. Words that occur in similar contexts have similar meanings.
- $\mathbf{A}$ is sparse and noisy; LSI/A "fills in" the zeroes and tries to eliminate the noise.

Probabilistic Topic Models

- LSI/A: assumes the elements of $\mathbf{A}$ are the result of Gaussian noise.
- Probabilistic Latent Semantic Analysis (PLSA) (Hofmann (1999)) model the probability distribution $p(\mathbf{x}_d|d)$
  - This is a particular kind of conditional language model
- Latent Dirichlet Allocation (Blei et al. (2003))
  - Introduce Bayesian inference to PLSA

Overview

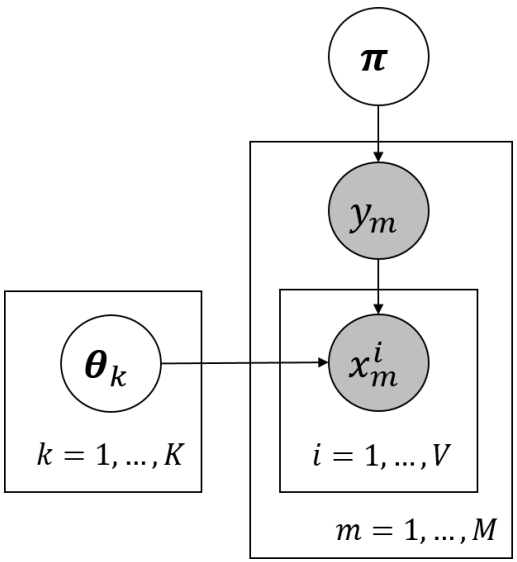
Document as a Sample of Mixed Topics



[Criticism of government response to the hurricane primarily consisted of criticism of its response to the approach of the storm and its aftermath, specifically in the delayed response]* to the *[flooding of New Orleans. ... 80% of the 1.3 million residents of the greater New Orleans metropolitan area evacuated] ...[Over seventy countries pledged monetary donations or other assistance]. ...

Mixture Models

- Recall naive Bayes based mixture models for a document collection by K topics (classes)
- Each topic is a multinomial over words, and each document is generated from a single topic



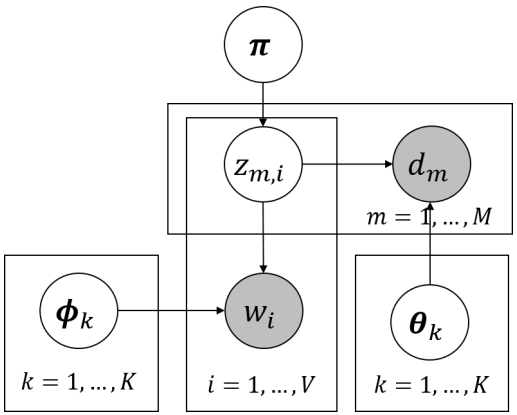
Naive Bayes from Class Conditional Unigram Model

- For $m = 1, \ldots, M$
  - Choose $y_m \sim Multinomial(y_m|1, \boldsymbol{\pi})$
  - Choose $N_m = \sum_j^d x_m^j \sim Poisson(\xi)$
  - For $n = 1, \ldots, N_m$
    - Choose $v \sim Multinomial(v|1, \boldsymbol{\theta}_{*|y_m}) = \prod_{j=1}^d (\theta_{*|y_m}^j)^{v=j}$

It assumes "one document, one topic."

Probabilistic Latent Semantic Analysis (PLSA)

- PLSA assumes that each document d (with word vector w) is generated from all topics, with documentspecific topic weights.



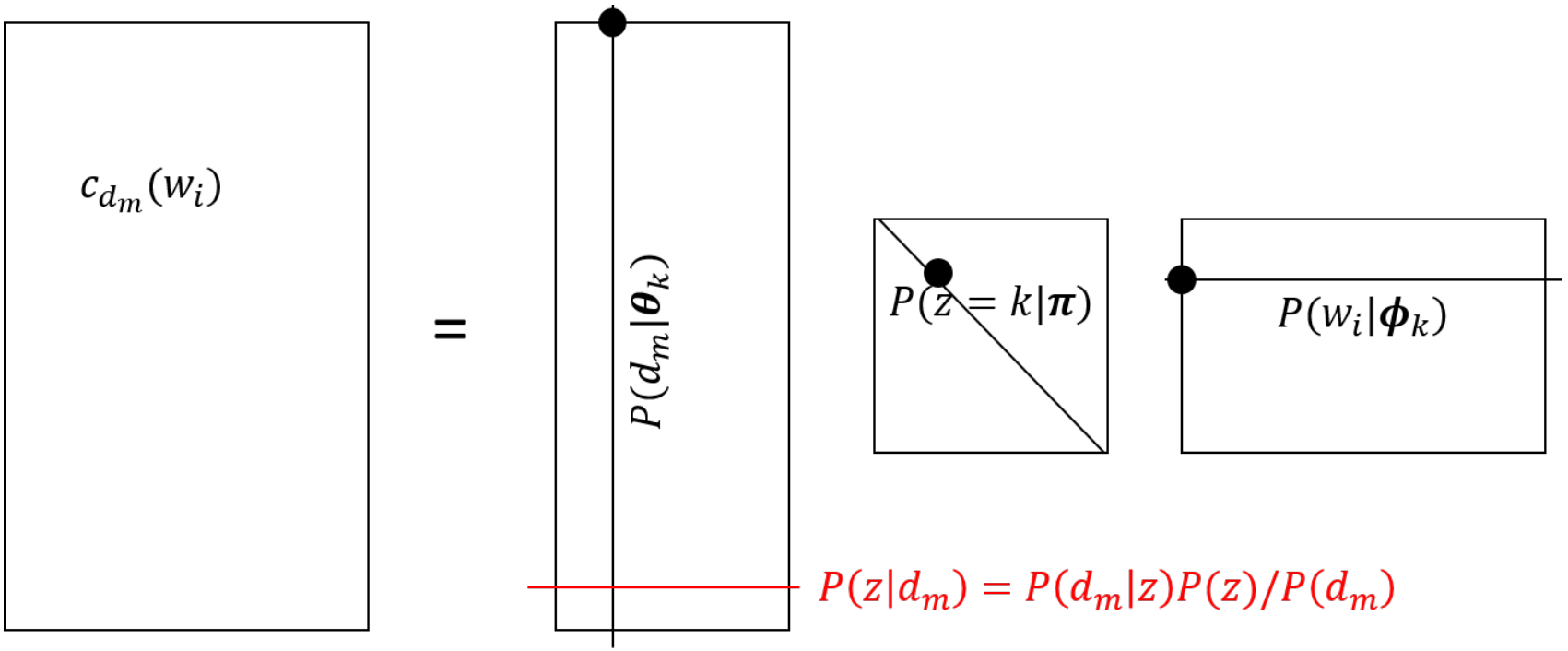
  - Choose a $z_{m,i} = k$ from topic distribution $\boldsymbol{\pi}$
  - Choose a document from $d_m \sim Multinomial(d_m|1, \boldsymbol{\theta}_k)$
  - Choose a word w_i from $w_i \sim Multinomial(w_i|1, \phi_k)$
  - Add one count of word w_i to document d_m
  - Repeat until we generate the document-word matrix

Under this process, the probability of picking the corpus is:

$$\begin{aligned} P(\mathcal{D}, \mathcal{W}) &= \prod_{m=1}^{M} \prod_{i=1}^{N_m} \sum_{k=1}^{K} P(z_{m,i} = k) P(d_m|\boldsymbol{\theta}_k) P(w_i|\phi_k) \\ &= \prod_{m=1}^{M} \prod_{i=1}^{V} \left(\sum_{k=1}^{K} P(z_{m,i} = k) P(d_m|\boldsymbol{\theta}_k) P(w_i|\phi_k) \right)^{c_{d_m}(w_i)} \end{aligned}$$

A Matrix Factorization View

$$
\begin{aligned}
P(\mathcal{D}, \mathcal{W}) &= \prod_{m=1}^{M} \prod_{i=1}^{N_m} \sum_{k=1}^{K} P(z_{m,i} = k) P(d_m|\boldsymbol{\theta}_k) P(w_i|\boldsymbol{\phi}_k) \\
&= \prod_{m=1}^{M} \prod_{i=1}^{V} \left(\sum_{k=1}^{K} P(z_{m,i} = k) P(d_m|\boldsymbol{\theta}_k) P(w_i|\boldsymbol{\phi}_k) \right)^{c_{d_m}(w_i)}
\end{aligned}
$$



Maximize Log Likelihood

- Log likelihood:

$$P(\mathcal{D}, \mathcal{W}) = \prod_{m=1}^{M} \prod_{i=1}^{V} \left(\sum_{k=1}^{K} P(z_{m,i} = k) P(d_m | \boldsymbol{\theta}_k) P(w_i | \boldsymbol{\phi}_k) \right)^{c_{d_m}(w_i)}$$

- To reduce the notation complexity, we denote:

$$\log P(\mathcal{D}, \mathcal{W}) = \sum_{d=1}^{M} \sum_{w=1}^{V} c_d(w) \log \left(\sum_{k=1}^{K} P(z) P(d|z) P(w|z) \right)$$

- We denote the parameters as
  $\Theta = \{\boldsymbol{\pi}, \boldsymbol{\phi}_k, \boldsymbol{\theta}_k, k = 1, \ldots, K\} = \{P(z), P(d|z), P(w|z)\}$
- Note here z is a hidden variable, and note that the sum is inside the log
- We can apply EM algorithm to maximize the likelihood

Lower Bound and E-Step

- Remember Jensens inequality

$$\log \sum_i P_i f_i(x) \geq \sum_i P_i \log f_i(x)$$

- We first compute the lower bound of the log likelihood:

$$\begin{aligned}
& \log P(\mathcal{D}, \mathcal{W}) \\
= & \sum_{d=1}^{M} \sum_{w=1}^{V} c_d(w) \log \left(\sum_{k=1}^{K} P(z) P(d|z) P(w|z) \right) \\
= & \sum_{d=1}^{M} \sum_{w=1}^{V} c_d(w) \log \left(\sum_{k=1}^{K} q_{z,d,w}(\Theta) \frac{P(z) P(d|z) P(w|z)}{q_{z,d,w}(\Theta)} \right) \\
\geq & \sum_{d=1}^{M} \sum_{w=1}^{V} c_d(w) \sum_{k=1}^{K} q_{z,d,w}(\Theta) \left(\log \frac{P(z) P(d|z) P(w|z)}{q_{z,d,w}(\Theta)} \right)
\end{aligned}$$

- Note Jensen's inequality involves computing $q_{z,d,w}(\Theta) = P(z|d, w, \Theta^t)$, which computes the probability of topics separately for each cell, under the current parameters Θ^t
- This is exactly the E-step:

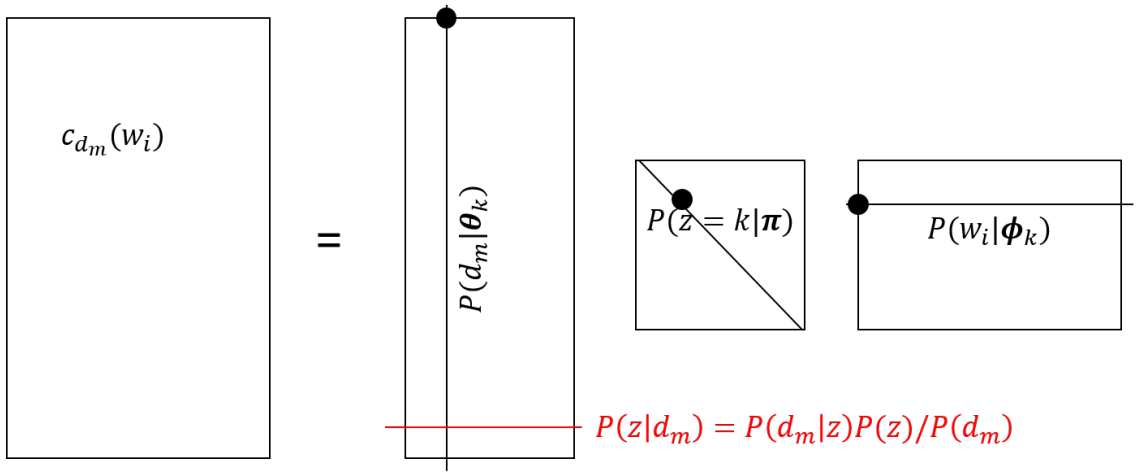
$$P(z|d, w, \Theta^t) \propto P(z|\Theta^t) P(d|z, \Theta^t) P(w|z, \Theta^t)$$

M-Step

$$
\begin{aligned}
& \log P(\mathcal{D}, \mathcal{W}) \\
= & \sum_{d=1}^{M} \sum_{w=1}^{V} c_d(w) \log \left(\sum_{k=1}^{K} P(z) P(d|z) P(w|z) \right) \\
= & \sum_{d=1}^{M} \sum_{w=1}^{V} c_d(w) \log \left(\sum_{k=1}^{K} P(z|d, w, \Theta^t) \frac{P(z) P(d|z) P(w|z)}{P(z|d,w,\Theta^t)} \right) \\
= & \sum_{d=1}^{M} \sum_{w=1}^{V} c_d(w) \sum_{k=1}^{K} P(z|d, w, \Theta^t) \left(\log \frac{P(z) P(d|z) P(w|z)}{P(z|d,w,\Theta^t)} \right)
\end{aligned}
$$

- Maximizing the right of the above inequality by setting the gradient to zero amounts to the M-step, which gives
  - $P(z) \propto \sum_d \sum_w c_d(w) P(z|d, w, \Theta^t)$
  - $P(d|z) \propto \sum_w c_d(w) P(z|d, w, \Theta^t)$
  - $P(w|z) \propto \sum_d c_d(w) P(z|d, w, \Theta^t)$

Use of PLSA



- Once the model is trained, we can look at it in the following way
  - $P(w|z)$ are the topics. Each topic is defined by a word multinomial. Often people find that the topics seem to have distinct semantic meanings.
  - From $P(d|z)$ and $P(z)$, we can compute $P(z|d) \propto p(d|z)p(z)$. $P(z|d)$ is the topic wights for document d.
- One drawback of PLSA is that it is transductive in nature. That is, there is no easy way to handle a new document that is not already in the collection
- This motivates us to introduce a Bayesian modeling of topic models

Illustration of EM

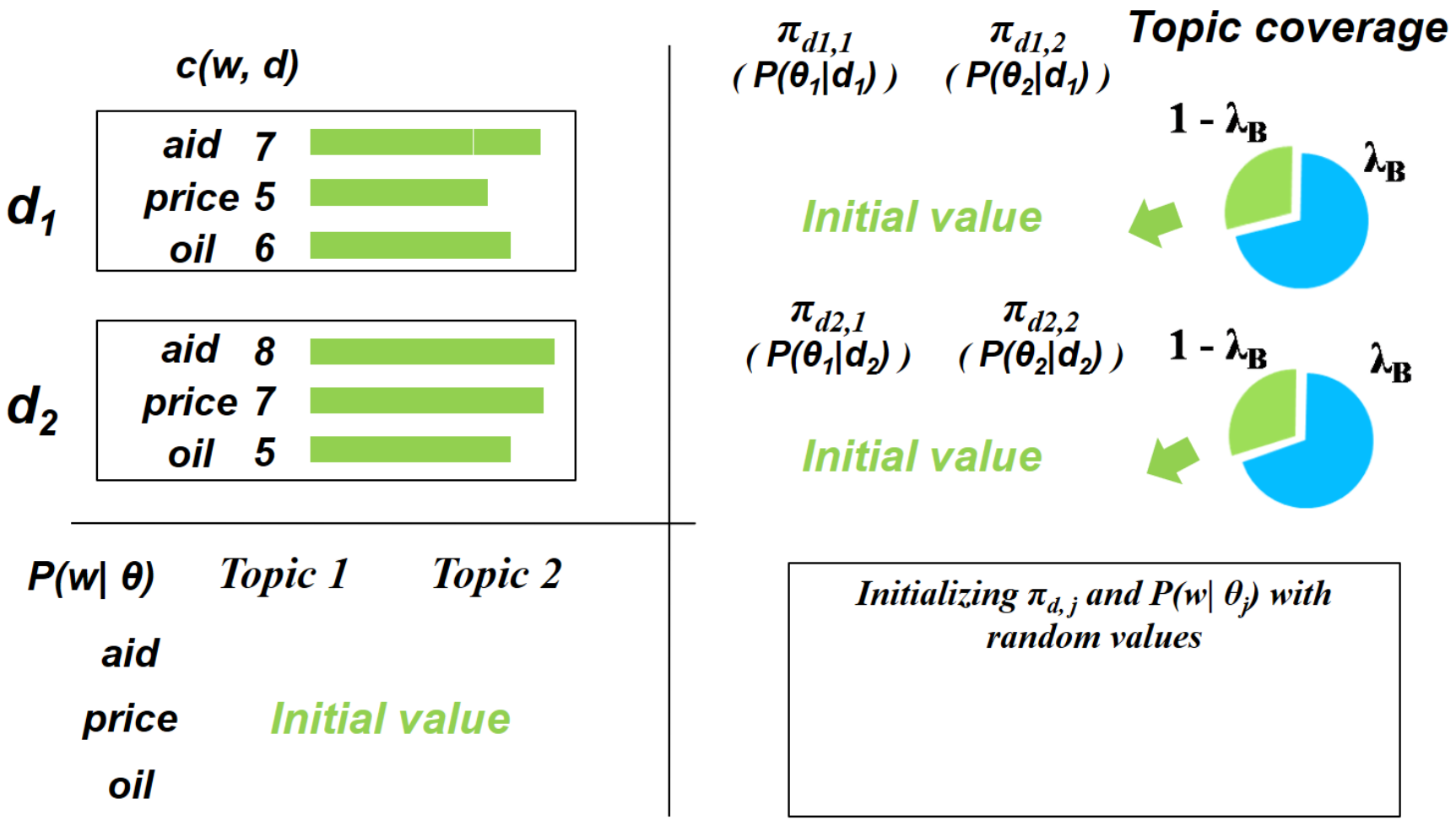


Illustration of EM

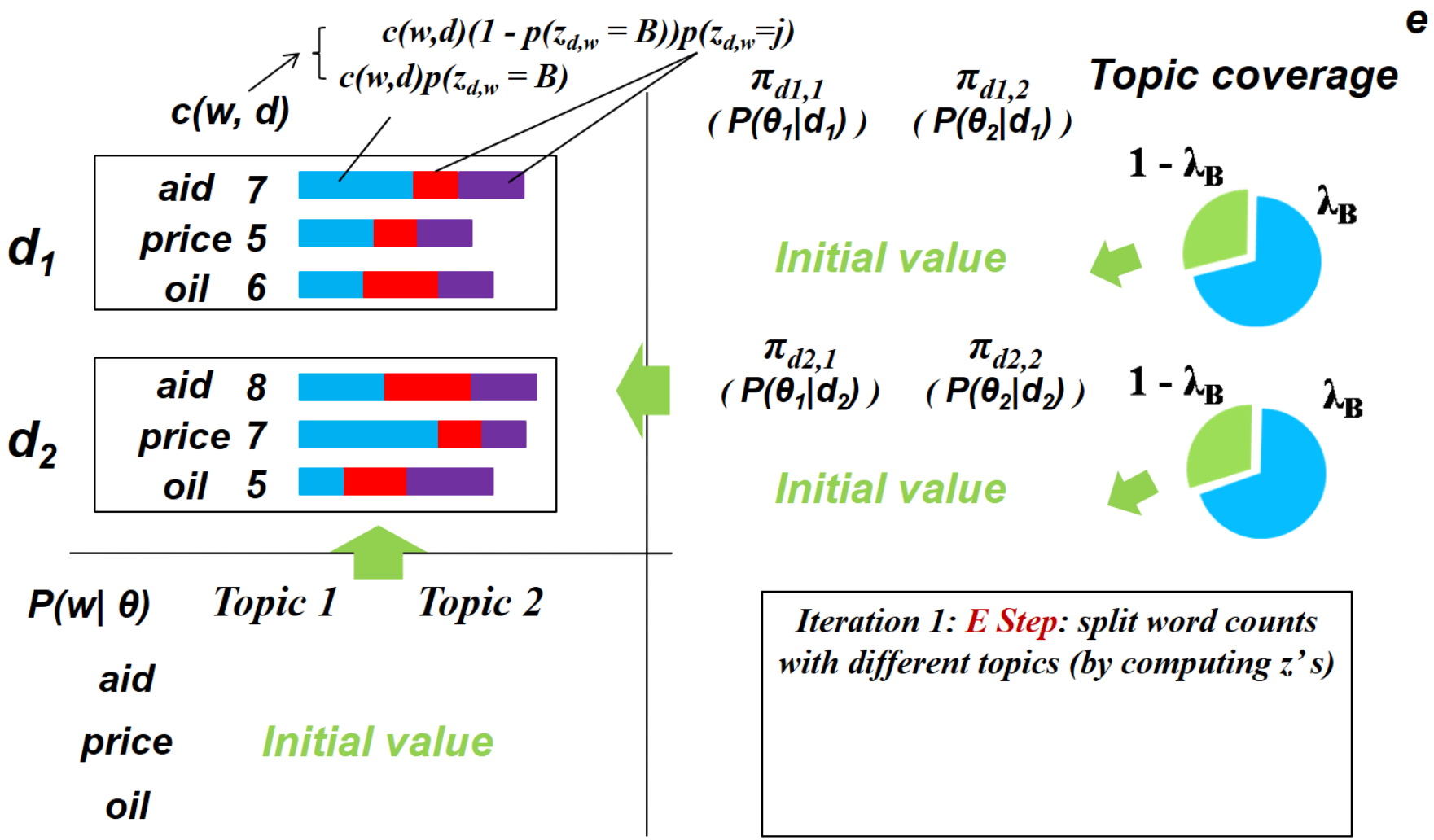


Illustration of EM

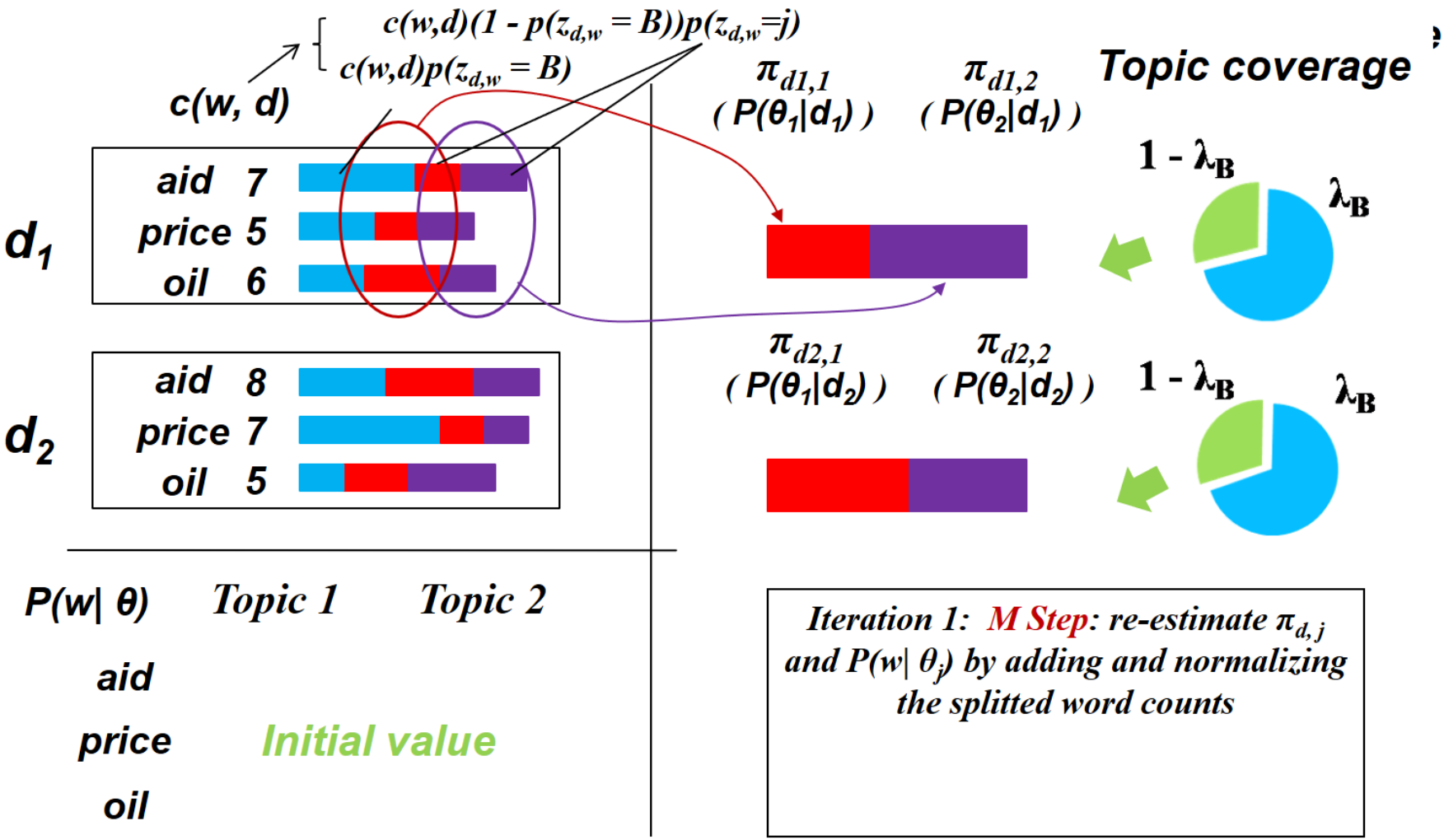


Illustration of EM

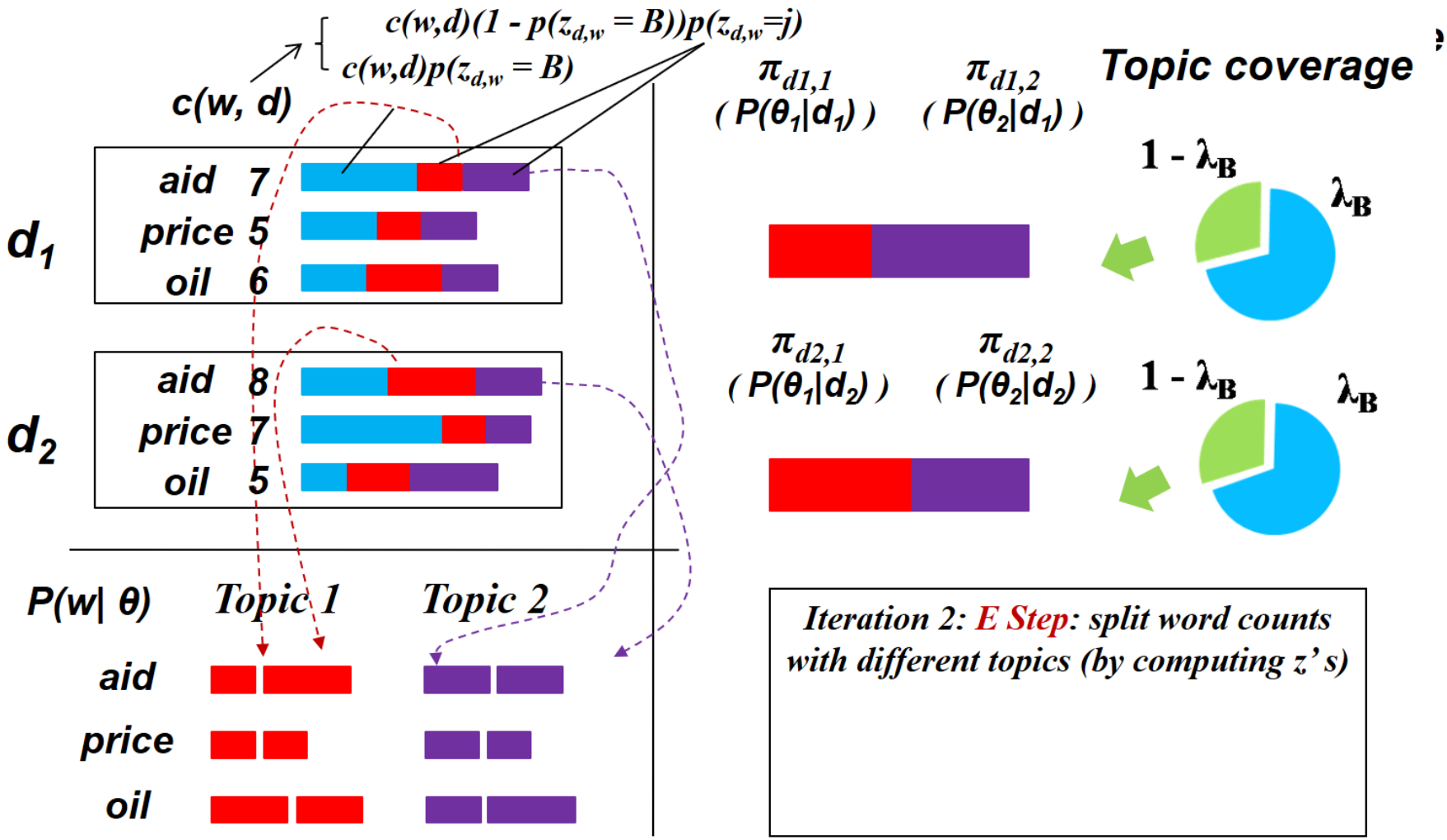


Illustration of EM

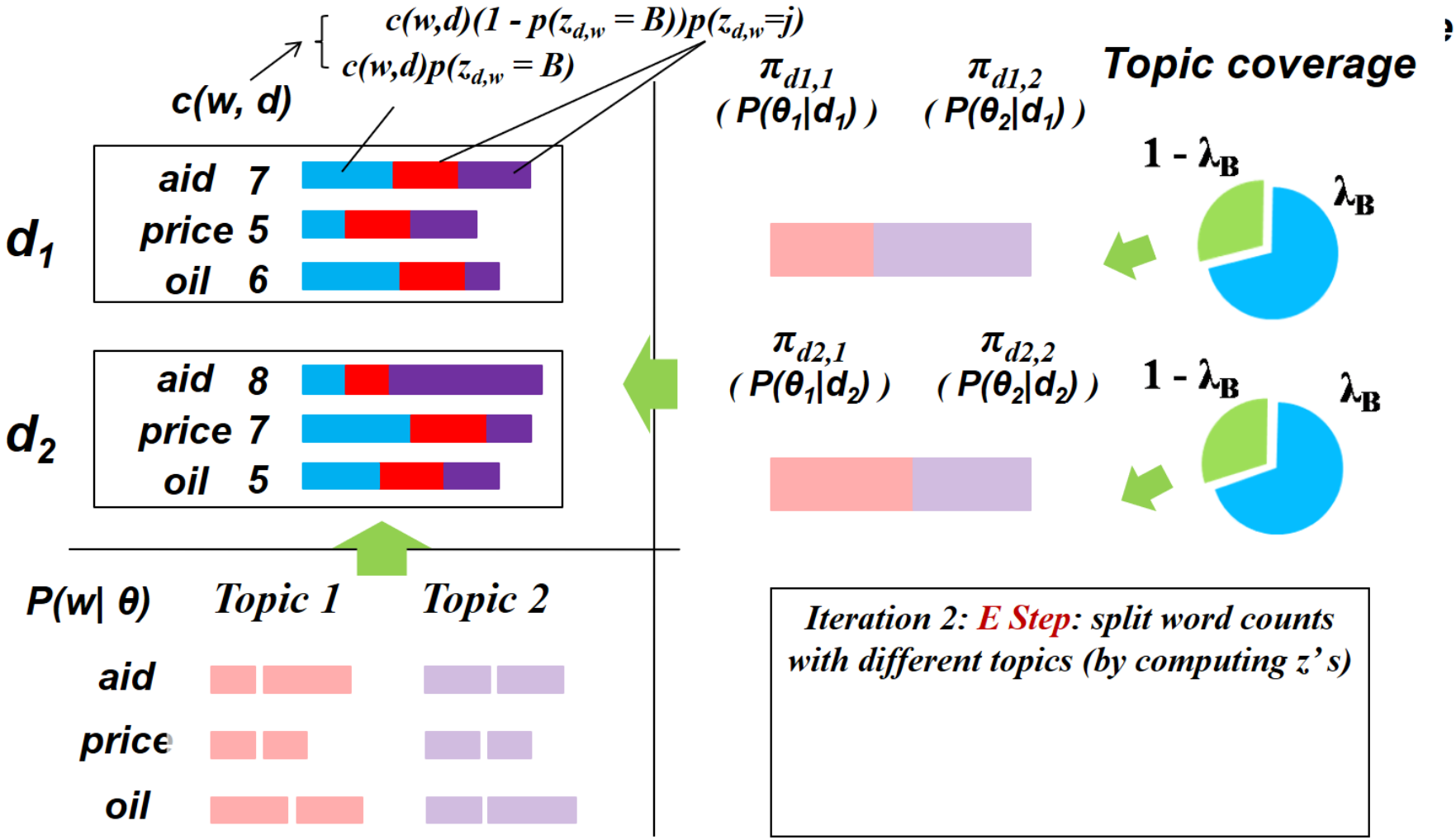
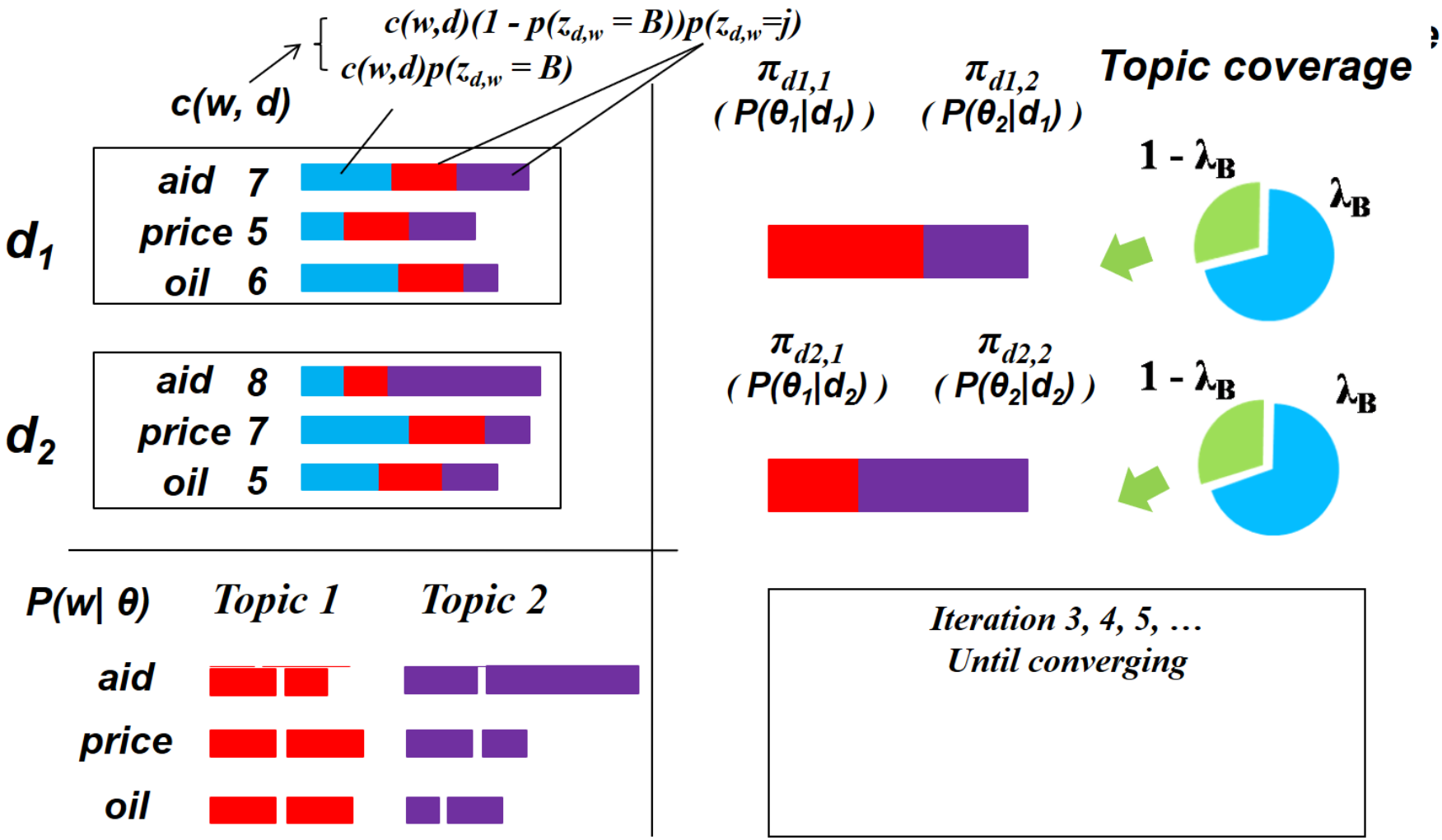


Illustration of EM



Use of Topic Models

"Arts"	"Budgets"	"Children"	"Education"
NEW	MILLION	CHILDREN	SCHOOL
FILM	TAX	WOMEN	STUDENTS
SHOW	PROGRAM	PEOPLE	SCHOOLS
MUSIC	BUDGET	CHILD	EDUCATION
MOVIE	BILLION	YEARS	TEACHERS
PLAY	FEDERAL	FAMILIES	HIGH
MUSICAL	YEAR	WORK	PUBLIC
BEST	SPENDING	PARENTS	TEACHER
ACTOR	NEW	SAYS	BENNETT
FIRST	STATE	FAMILY	MANIGAT
YORK	PLAN	WELFARE	NAMPHY
OPERA	MONEY	MEN	STATE
THEATER	PROGRAMS	PERCENT	PRESIDENT
ACTRESS	GOVERNMENT	CARE	ELEMENTARY
LOVE	CONGRESS	LIFE	HAITI

The William Randolph Hearst Foundation will give $1.25 million to Lincoln Center, Metropolitan Opera Co., New York Philharmonic and Juilliard School. "Our board felt that we had a real opportunity to make a mark on the future of the performing arts with these grants an act every bit as important as our traditional areas of support in health, medical research, education and the social services," Hearst Foundation President Randolph A. Hearst said Monday in announcing the grants. Lincoln Center's share will be $200,000 for its new building, which will house young artists and provide new public facilities. The Metropolitan Opera Co. and New York Philharmonic will receive $400,000 each. The Juilliard School, where music and the performing arts are taught, will get $250,000. The Hearst Foundation, a leading supporter of the Lincoln Center Consolidated Corporate Fund, will make its usual annual $100,000 donation, too.

Example of topics found from a Science Magazine papers collection

universe	0.0439
galaxies	0.0375
clusters	0.0279
matter	0.0233
galaxy	0.0232
cluster	0.0214
cosmic	0.0137
dark	0.0131
light	0.0109
density	0.01

drug	0.0672
patients	0.0493
drugs	0.0444
clinical	0.0346
treatment	0.028
trials	0.0277
therapy	0.0213
trial	0.0164
disease	0.0157
medical	0.00997

cells	0.0675
stem	0.0478
human	0.0421
cell	0.0309
gene	0.025
tissue	0.0185
cloning	0.0169
transfer	0.0155
blood	0.0113
embryos	0.0111

sequence	0.0818
sequences	0.0493
genome	0.033
dna	0.0257
sequencing	0.0172
map	0.0123
genes	0.0122
chromosome	0.0119
regions	0.0119
human	0.0111

years	0.156
million	0.0556
ago	0.045
time	0.0317
age	0.0243
year	0.024
record	0.0238
early	0.0233
billion	0.0177
history	0.0148

bacteria	0.0983
bacterial	0.0561
resistance	0.0431
coli	0.0381
strains	0.025
microbiol	0.0214
microbial	0.0196
strain	0.0165
salmonella	0.0163
resistant	0.0145

male	0.0558
females	0.0541
female	0.0529
males	0.0477
sex	0.0339
reproductive	0.0172
offspring	0.0168
sexual	0.0166
reproduction	0.0143
eggs	0.0138

theory	0.0811
physics	0.0782
physicists	0.0146
einstein	0.0142
university	0.013
gravity	0.013
black	0.0127
theories	0.01
aps	0.00987
matter	0.00954

immune	0.0909
response	0.0375
system	0.0358
responses	0.0322
antigen	0.0263
antigens	0.0184
immunity	0.0176
immunology	0.0145
antibody	0.014
autoimmune	0.0128

stars	0.0524
star	0.0458
astrophys	0.0237
mass	0.021
disk	0.0173
black	0.0161
gas	0.0149
stellar	0.0127
astron	0.0125
hole	0.00824

Remarks

- A document is now characterized as a mixture of corpus-universal topics (each of which is a unigram model).
- Topic mixtures can be incorporated into language models; see Iyer and Ostendorf (1999), for example.
- Compared to LSI/A: PLSA is more interpretable (e.g., LSI/A can give negative values!).
- PLSA cannot assign probability to a text not in $\mathcal{W}$; it only defines conditional distributions over words given texts in $\mathcal{W}$.
- The next model overcomes this problem by adding another level of randomness: $P(z|d)$ becomes a random variable, not a parameter.

Overview

Recall Unigram Language Modeling

- Data corpus: a collection of words, $\mathcal{W} = \{w_1, w_2, \ldots, w_N\}$
- Model: multinomial distribution $P(\mathcal{W}|\boldsymbol{\theta})$ with parameters $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_V)$, where
  - $\theta_i = P(v_i)$
  - $v_i \in \mathcal{V}$
  - $\mathcal{V}$ is the vocabulary
  - $|\mathcal{V}| = V$
- Count of words in corpus $\mathbf{u} = (u_1, \ldots, u_V)$ where $u_i = c(v_i)$ is the count of v_i shown in $\mathcal{W}$, $\sum_i u_i = N$

Unigram Modeling

- "Bag of words" assumes the words are sampled from a multinomial distribution $\mathbf{u} \sim \text{Multi}(\boldsymbol{\theta})$

$$P(\mathbf{u}|\boldsymbol{\theta}) = \begin{pmatrix} N \\ \mathbf{u} \end{pmatrix} \prod_{i=1}^{V} \theta_i^{u_i} \triangleq \text{Mult}(\mathbf{u}|\boldsymbol{\theta}, N), \textit{where} \begin{pmatrix} N \\ \mathbf{u} \end{pmatrix} = \frac{N!}{\prod_i u_i!}$$

If we focus on a single trial, we have:

$$P(w|\boldsymbol{\theta}) = P(w = v_i) = \prod_{i=1}^{V} \theta_i^{\delta_{w=v_i}} \triangleq \text{Mult}(w|\boldsymbol{\theta})$$

- Maximum likelihood estimator: $\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} P(\mathcal{W}|\boldsymbol{\theta})$

$$P(\mathcal{W}|\boldsymbol{\theta}) = \prod_{j=1}^{N} P(w_j|\boldsymbol{\theta}) = \prod_{i=1}^{V} P(v_i)^{u_i} = \prod_{i=1}^{V} \theta^{u_i}$$

Maximum Likelihood Estimation: $\hat{\theta} = \arg\max_{\boldsymbol{\theta}} P(\mathcal{W}|\boldsymbol{\theta})$

$$P(\mathcal{W}|\boldsymbol{\theta}) = \prod_i^V \theta_i^{u_i}$$

(log likelihood)

$$\Rightarrow \log P(\mathcal{W}|\boldsymbol{\theta}) = \sum_i^V u_i \log \theta_i$$

(Lagrange multiplier to make $\boldsymbol{\theta}$ be a distribution)

$$\Rightarrow L(\mathcal{W}, \boldsymbol{\theta}) = \log P(\mathcal{W}|\boldsymbol{\theta}) = \sum_i^V u_i \log \theta_i + \lambda(\sum_i \theta_i - 1)$$

(Set partial derivatives to zero)

$$\Rightarrow \frac{\partial L}{\partial \theta_i} = \frac{u_i}{\theta_i} + \lambda$$

Since $\sum_i^V \theta_i = 1$, we have $\lambda = -\sum_i^V u_i$

$$\Rightarrow \theta_i = \frac{u_i}{\sum_i^V u_i} = \frac{u_i}{N} \ (\textit{Maximum Likelihood Estimation}, \textit{MLE})$$

Generalization: Add-K smoothing

Problem: Add-one moves too much probability mass from seen to unseen events!

- Variant of Add-One smoothing
  - Add a constant k to the counts of each word
  - For any $k > 0$ (typically, $k < 1$), a unigram model is

$$\Rightarrow \theta_i = \frac{u_i + k}{\sum_i^V u_i + kV} = \frac{u_i + k}{N + kV}$$

- If $k = 1$
  - "Add one" Laplace smoothing
- This is still too simplistic to work well.

Any explanation?

Bayesian Interpretation

- Conjugate distribution
  - Adding a conjugate prior to a likelihood will result in a posterior in the same distribution family as the prior, then the prior and the likelihood are called conjugate distributions
  - Conjugate distribution makes us easier to formulate Bayesian belief and inference the model

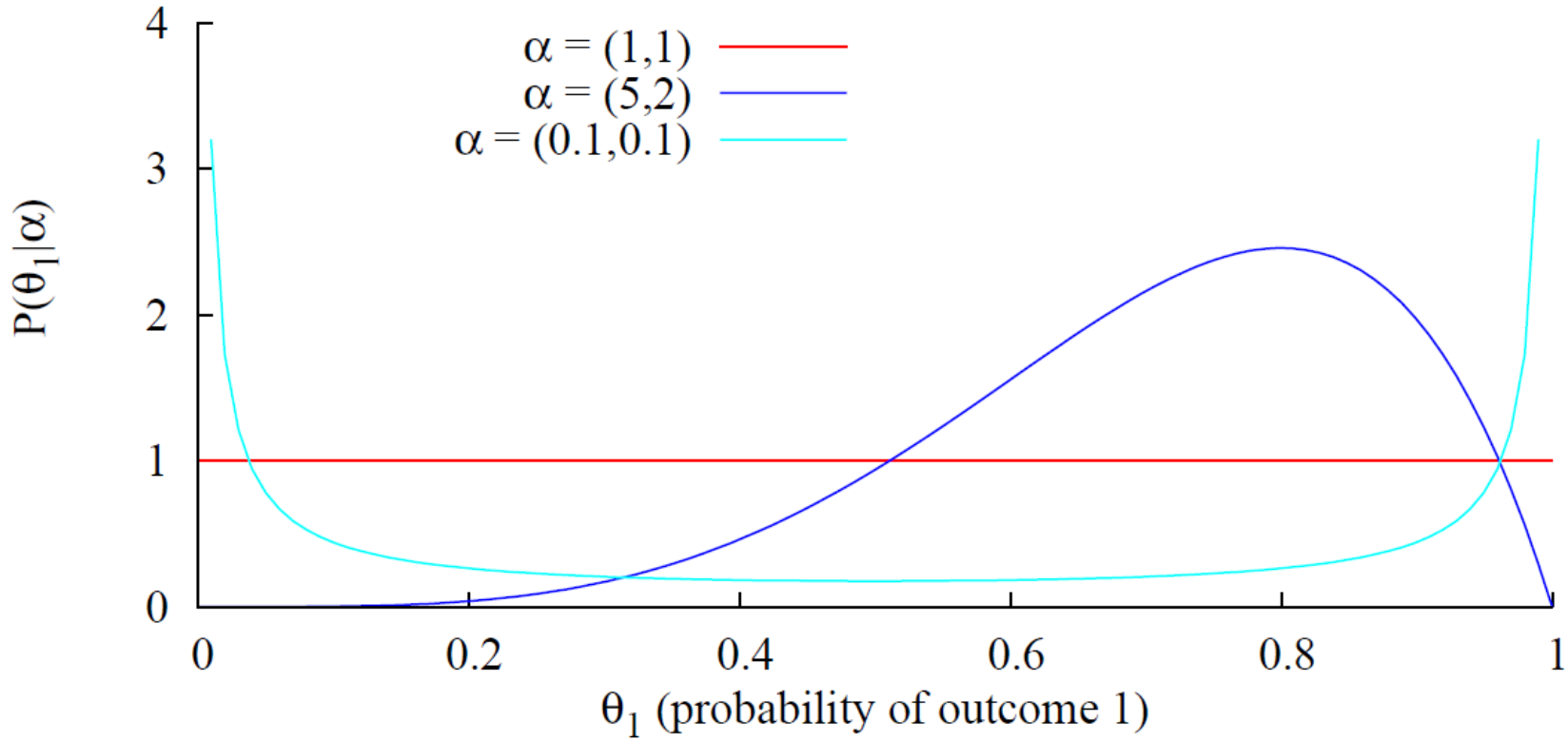
Bayesian Interpretation

- The conjugate prior of a multinomial is Dirichlet distribution:
  $P(\boldsymbol{\theta}|\boldsymbol{\alpha}) = \mathrm{Dir}(\boldsymbol{\theta}|\boldsymbol{\alpha}) \triangleq \frac{\Gamma(\sum_{i=1}^{V}\alpha_i)}{\prod_{i=1}^{V}\Gamma(\alpha_i)}\prod_{i=1}^{V}\theta_i^{\alpha_i-1} \triangleq \frac{1}{\Delta(\boldsymbol{\alpha})}\prod_{i=1}^{V}\theta_i^{\alpha_i-1}$
  - The "Dirichlet Delta function" $\Delta(\boldsymbol{\alpha})$ is introduced for convenience
  - $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_V)^\top \in \mathbb{R}^V$
  - The Gamma function satisfies $\Gamma(x+1) = x\Gamma(x)$
    - For integer variable, Gamma function is $\Gamma(x) = (x-1)!$
    - For real numbers, it is $\Gamma(x) = \int_0^\infty t^{x-1}e^{-t}\mathrm{d}t$
- The Dirichlet distribution can be seen as the *"distribution of a distribution"*
  - We can sample a multinomial distribution from Dirichlet distribution, satisfied the constraint $\sum_i \theta_i = 1$

Beta Distribution

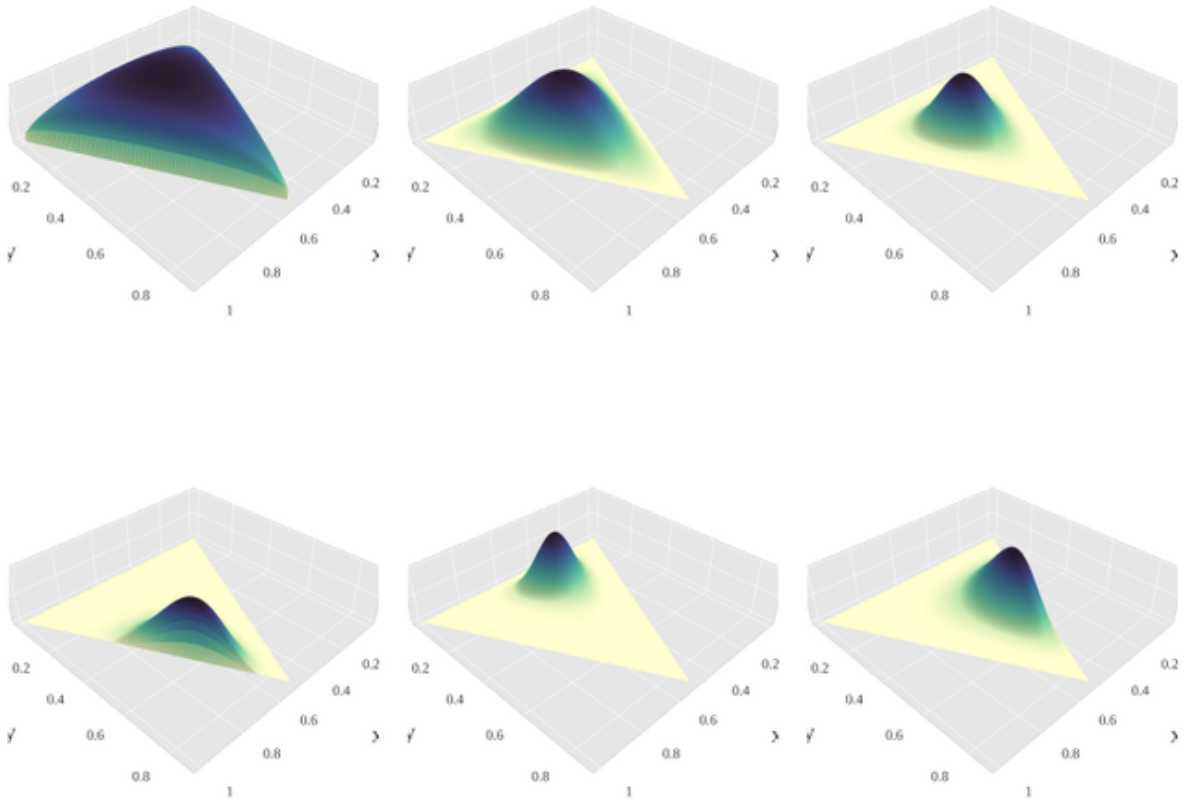
Called Beta distribution when there are two choices of variable values

$$P(\boldsymbol{\theta} \mid \boldsymbol{\alpha}) = \frac{\Gamma(\sum_{j=1}^{m} \alpha_j)}{\prod_{j=1}^{m} \Gamma(\alpha_j)} \prod_{k=1}^{m} \theta_k^{\alpha_k - 1}$$



Bayesian Interpretation

- The Dirichlet distribution can be seen as the *"distribution of a distribution"*
  - We can sample a multinomial distribution from Dirichlet distribution, satisfied the constraint $\sum_i \theta_i = 1$



Bayesian Estimation

- Remember Maximum likelihood estimator: $\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} P(\mathcal{W}|\boldsymbol{\theta})$

$$P(\mathcal{W}|\boldsymbol{\theta}) = \prod_{j=1}^{N} P(w_j|\boldsymbol{\theta}) = \prod_{i=1}^{V} P(v_i)^{u_i} = \prod_{i=1}^{V} \theta^{u_i} (\theta_i = \frac{u_i}{\sum_i^V u_i} = \frac{u_i}{N})$$

- The posterior of the parameters $\boldsymbol{\theta}$ based on the prior and the observation of N words:

$$\begin{aligned} P(\boldsymbol{\theta}|\mathcal{W}, \boldsymbol{\alpha}) &= \frac{P(\mathcal{W}|\boldsymbol{\theta})P(\boldsymbol{\theta}|\boldsymbol{\alpha})}{P(\mathcal{W}|\boldsymbol{\alpha})} = \frac{\prod_{i=1}^{N} P(w_i|\boldsymbol{\theta})P(\boldsymbol{\theta}|\boldsymbol{\alpha})}{\int_{\boldsymbol{\theta}} \prod_{i=1}^{N} P(w_i|\boldsymbol{\theta})P(\boldsymbol{\theta}|\boldsymbol{\alpha})\mathrm{d}\boldsymbol{\theta}} \\ &= \frac{\prod_{i=1}^{N} P(w_i|\boldsymbol{\theta})P(\boldsymbol{\theta}|\boldsymbol{\alpha})}{Z} \\ &= \frac{1}{Z} \prod_{i=1}^{V} \theta_i^{u_i} \frac{1}{\Delta(\boldsymbol{\alpha})} \prod_{i=1}^{V} \theta_i^{\alpha_i - 1} \\ &= \frac{1}{\Delta(\boldsymbol{\alpha}+\mathbf{u})} \prod_{i=1}^{V} \theta_i^{\alpha_i + u_i - 1} = \mathrm{Dir}(\boldsymbol{\theta}|\boldsymbol{\alpha} + \mathbf{u}) \end{aligned}$$

- According to the property of Dirichlet distribution, the posterior is with mean $\theta_i = \frac{u_i + \alpha_i}{\sum_i^V u_i + V\alpha_i}$ and mode $\theta_i = \frac{u_i + \alpha_i - 1}{\sum_i^V u_i + V(\alpha_i - 1)}$ (MAP, maximum a posterior estimation, estimation), and $\alpha_i = 1$ equals to MLE

Graphical Representation of Two Versions

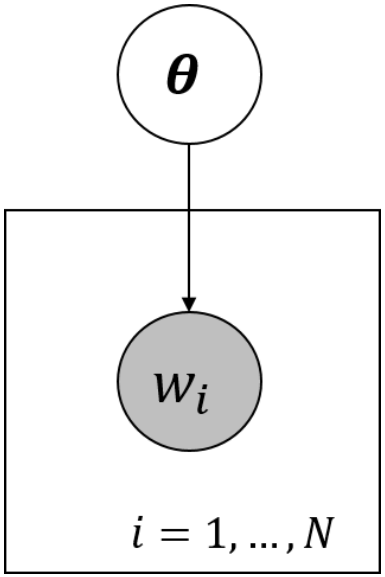


Figure: Unigram Language Model

$$P(\mathcal{W}|\boldsymbol{\theta}) = \prod_{j=1}^{N} P(w_j|\boldsymbol{\theta})$$

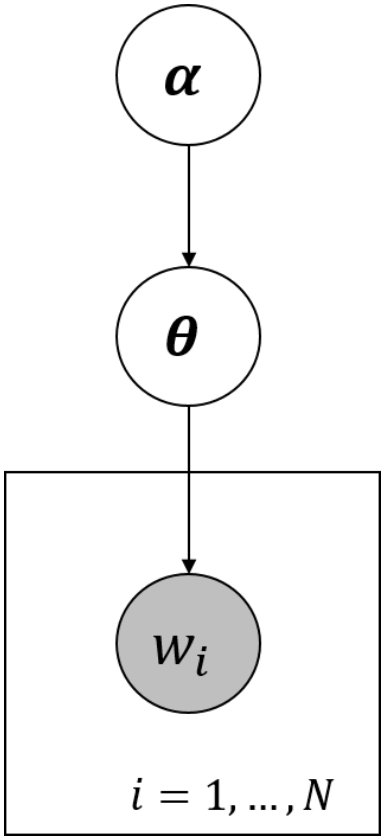
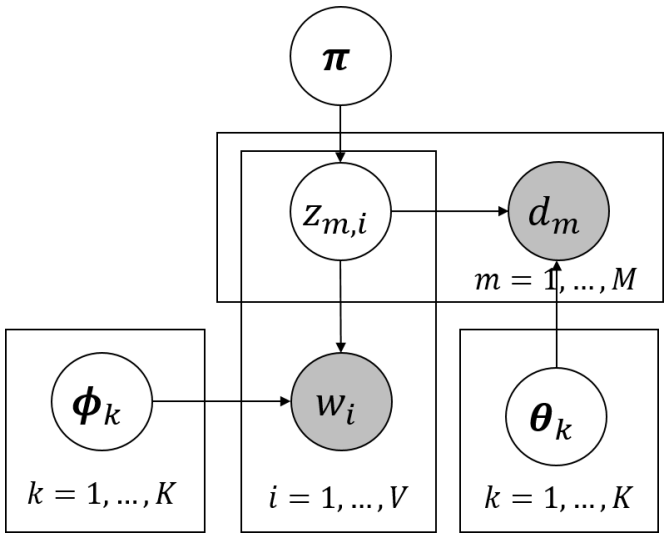


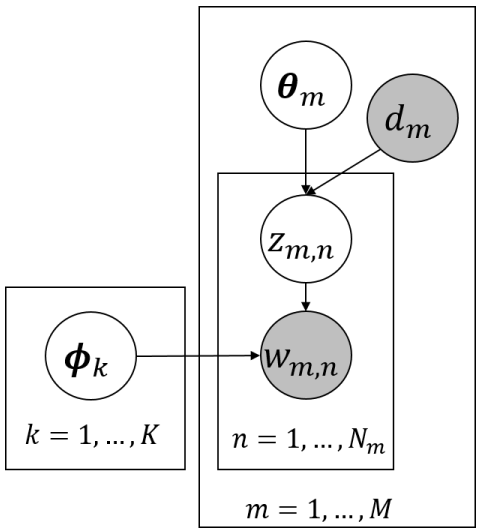
Figure: Bayesian Esitmation

$$P(\boldsymbol{\theta}|\mathcal{W}, \boldsymbol{\alpha}) = \frac{P(\mathcal{W}|\boldsymbol{\theta})P(\boldsymbol{\theta}|\boldsymbol{\alpha})}{P(\mathcal{W}|\boldsymbol{\alpha})}$$

Alternative Way for PLSA to Generate Texts

$$
\begin{aligned}
P(\mathcal{D}, \mathcal{W}) &= \prod_{m=1}^{M} \prod_{i=1}^{N_m} \sum_{k=1}^{K} P(z_{m,i} = k) P(d_m | \boldsymbol{\theta}_k) P(w_i | \boldsymbol{\phi}_k) \\
&= \prod_{m=1}^{M} \prod_{i=1}^{V} \left(\sum_{k=1}^{K} P(z_{m,i} = k) P(d_m | \boldsymbol{\theta}_k) P(w_i | \boldsymbol{\phi}_k) \right)^{c_{d_m}(w_i)}
\end{aligned}
$$

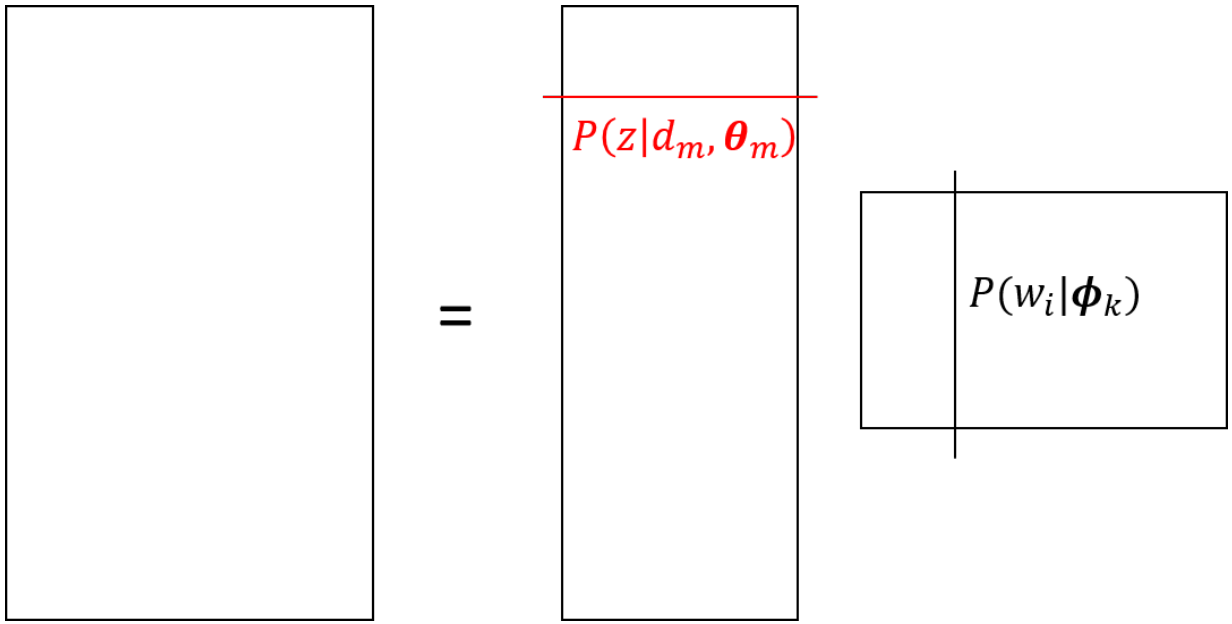




$$
\begin{aligned}
P(\mathcal{D}, \mathcal{W}) &= \prod_{m=1}^{M} \prod_{i=1}^{V} P(d_m) \left(\sum_{k=1}^{K} P(z_{m,i} = k | \boldsymbol{\theta}_m) P(w_i | \boldsymbol{\phi}_k) \right)^{c_{d_m}(w_i)}
\end{aligned}
$$

Comparison of Mixture Models and PLSA

$$P(\mathcal{D}, \mathcal{W}) = \prod_{m=1}^{M} \prod_{i=1}^{V} P(d_m) \left(\sum_{k=1}^{K} P(z_{m,i} = k|\boldsymbol{\theta}_m) P(w_i|\boldsymbol{\phi}_k) \right)^{c_{d_m}(w_i)}$$



Comparison of Mixture Models and PLSA

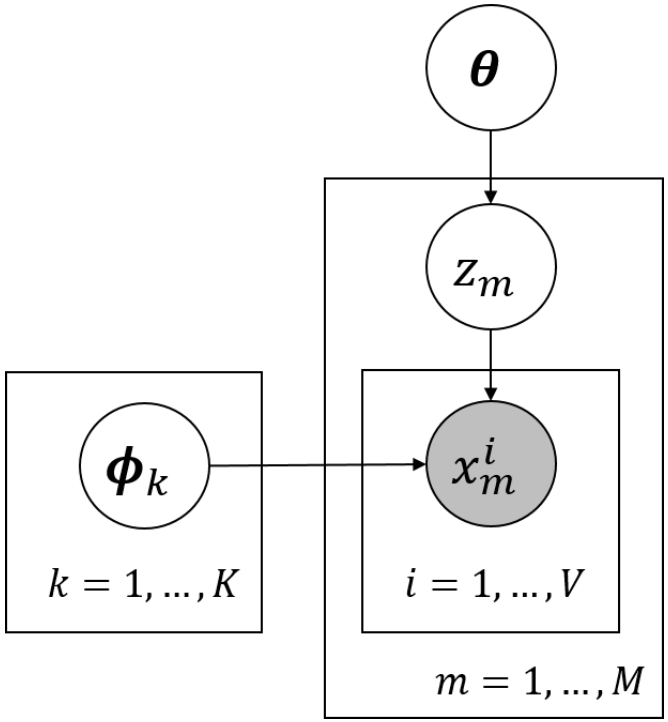


Figure: Mixture Models (with notation change)

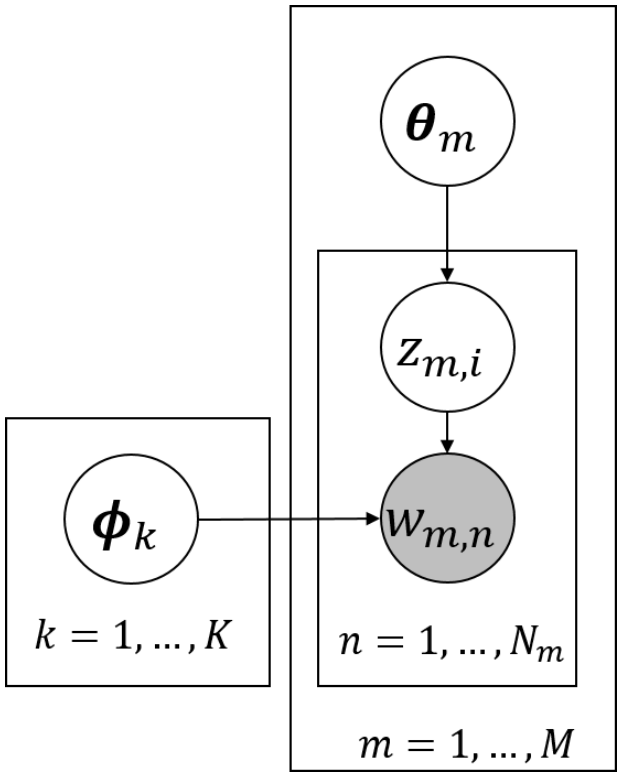


Figure: PLSA

Bayesian Modeling: Language Models

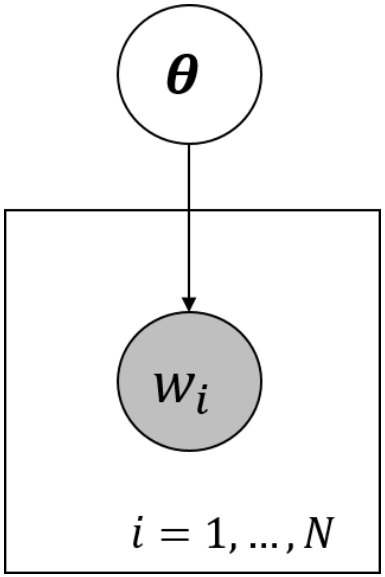


Figure: Unigram Language Model

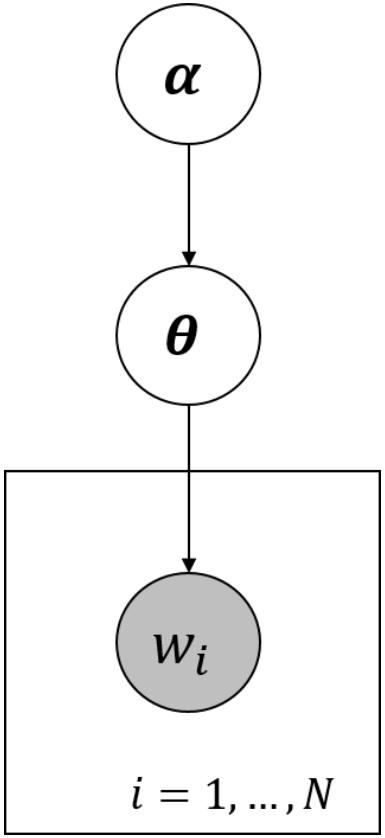


Figure: Bayesian Esitmation

Bayesian Modeling: Mixture Models

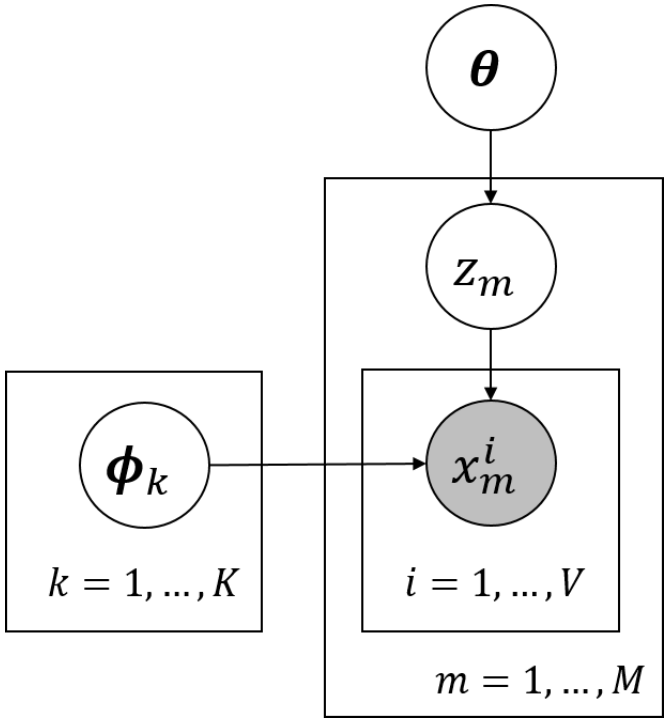


Figure: Unigram Language Model

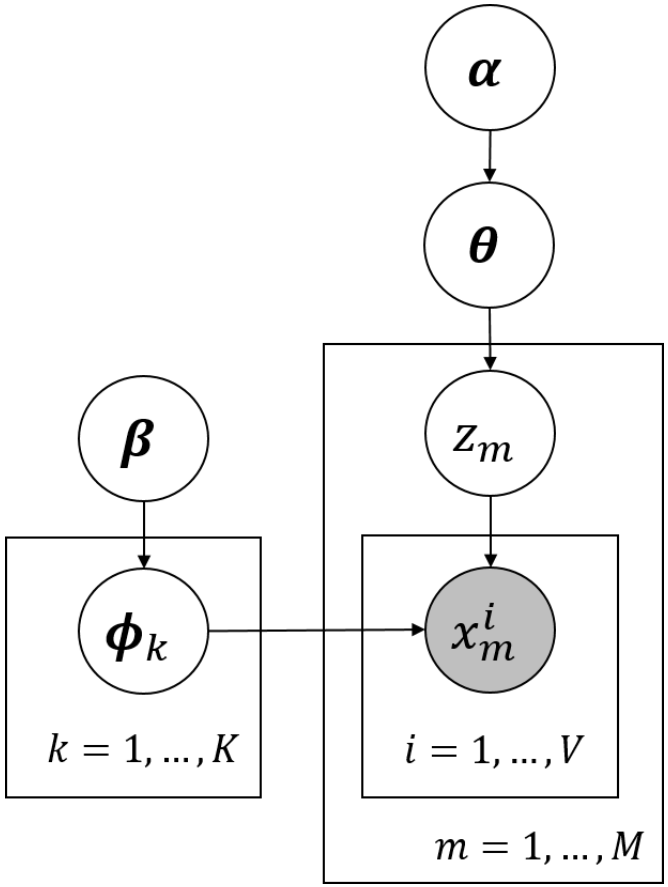


Figure: Bayesian Esitmation

Bayesian Modeling: Topic Models

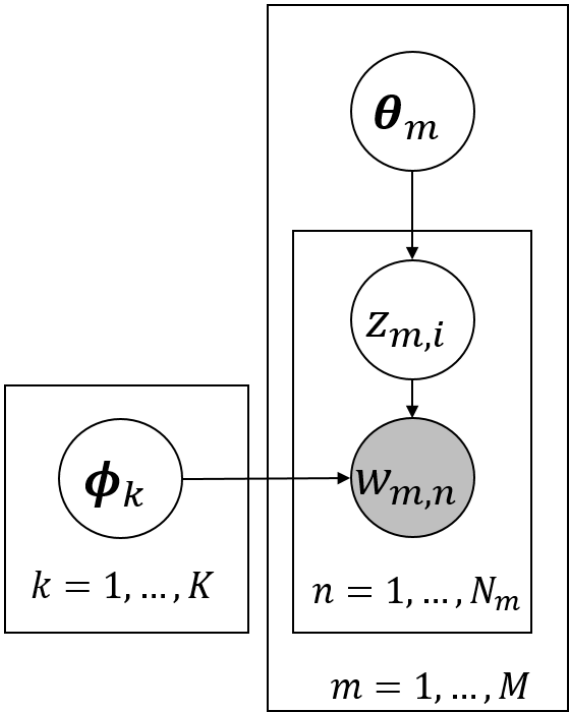


Figure: PLSA

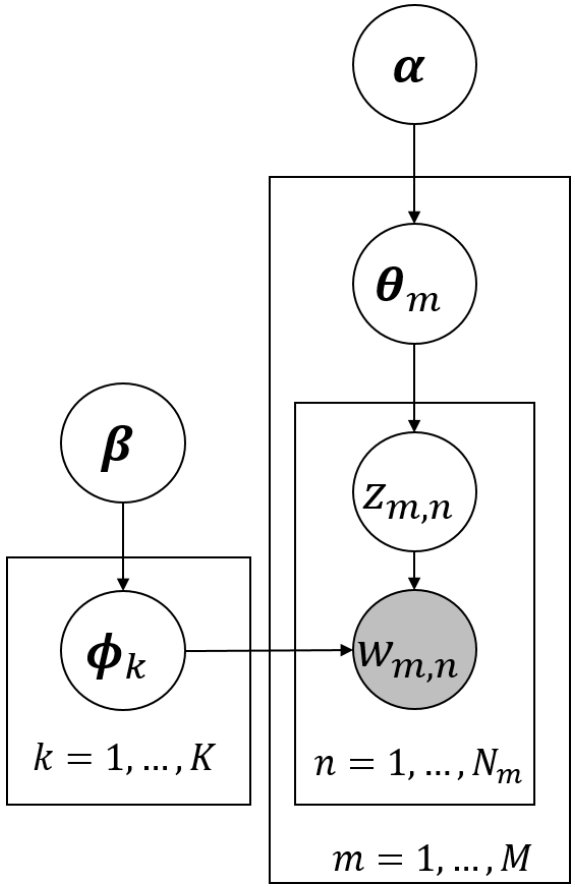


Figure: LDA

Generative Process of Latent Dirichlet Allocation

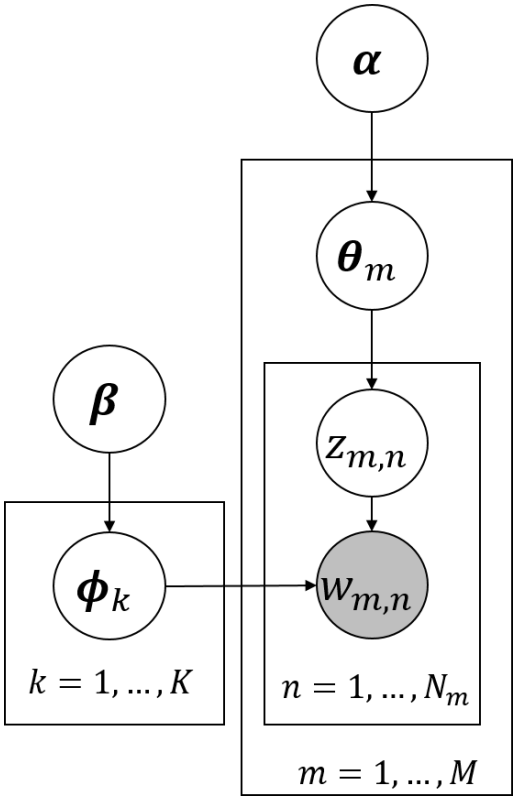


Figure: LDA

- For all clusters/components $k \in [1, K]$:
  - Choose mixture components $\phi_k \sim \mathrm{Dir}(\phi|\beta)$
- For all documents $m \in [1, M]$:
  - Choose $N_m \sim \mathrm{Poisson}(\xi)$
  - Choose mixture probability $\theta_m \sim \mathrm{Dir}(\theta|\alpha)$
  - For all words $n \in [1, N_m]$ in document d_m:
    - Choose a component index $z_{m,n} \sim \mathrm{Mult}(z|\theta_m)$
    - Choose a word $w_{m,n} \sim \mathrm{Mult}(w|\phi_{z_{m,n}})$

References I

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

Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., and Harshman, R. A. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science (JASIS)*, 41(6):391–407.

Firth, J. R. (1957). *A synopsis of linguistic theory 1930-55.*, volume 1952-59, pages 1–32. The Philological Society, Oxford.

Harris, Z. (1954). Distributional structure. *Word*, 10(23):146–162.

Hofmann, T. (1999). Probabilistic latent semantic analysis. In *UAI*, pages 289–296.

Salton, G., Wong, A., and Yang, C. (1975). A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620.