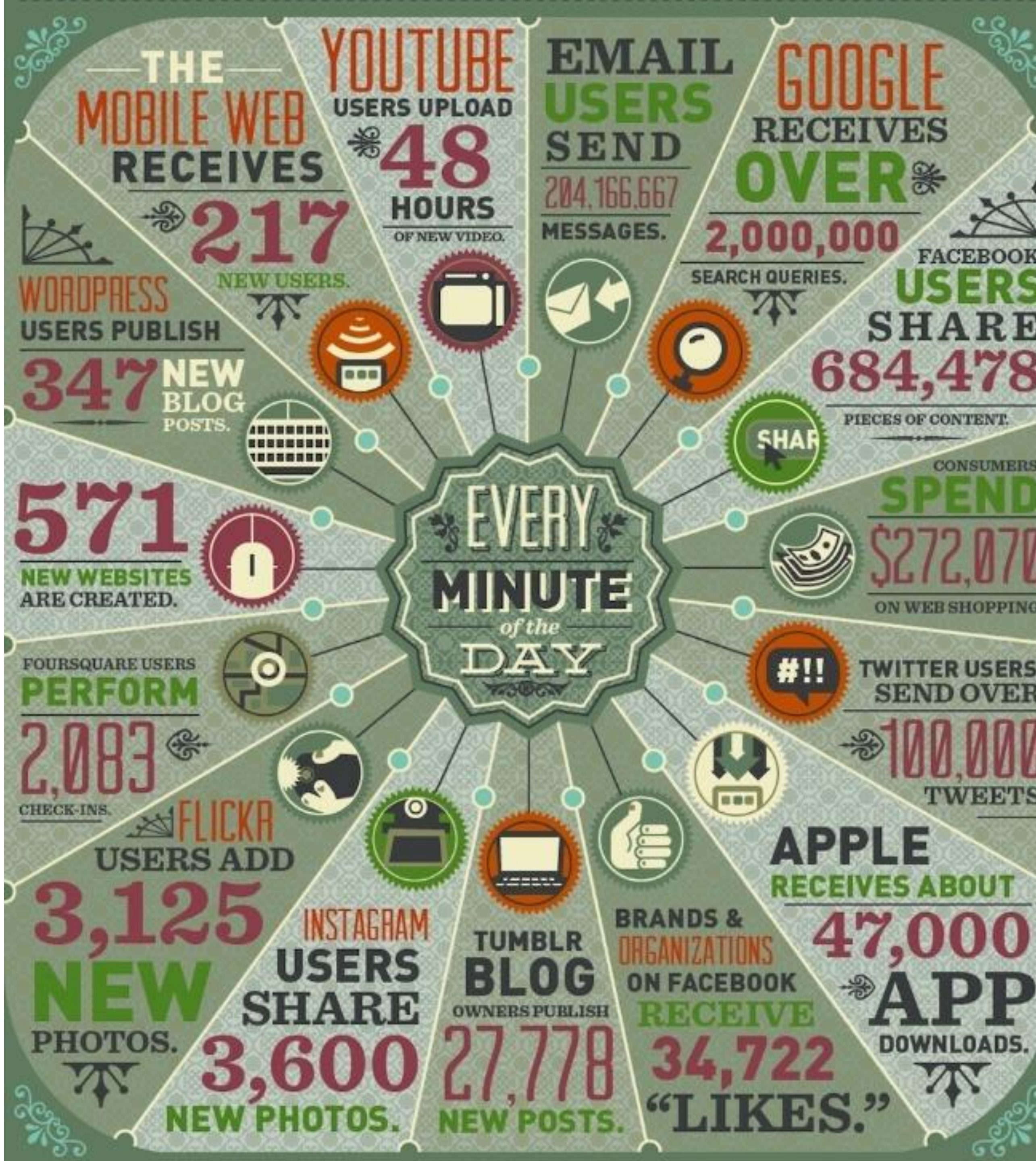
Topic Models

Why Topic Model

- Enormous amount of data is being generated every minute.
- Dimension reduction is necessary for effective use of the data.



User as Document

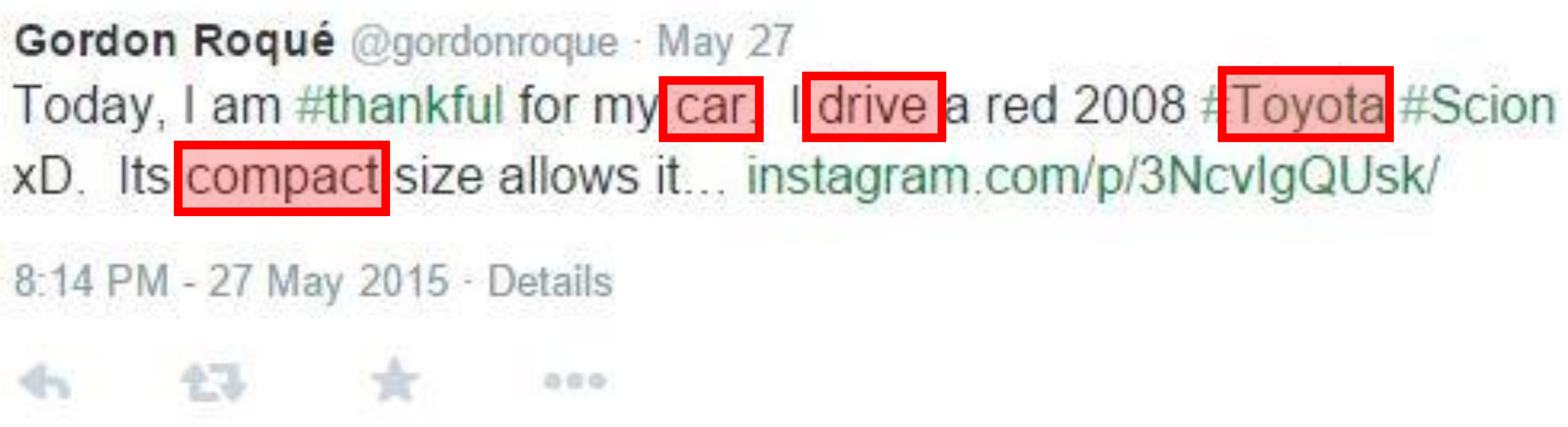
Consider that a company named M$ has collected user data from many sources. They are stored as documents (bags of words) under user ids.

Find Relevant User

Toyota wants to promote their new compact car by showing ads to users who are interested in cars.

M$ needs to extract users’ interested topics from:

- Search queries
- Tweets
- Posts
- And any sort of text records.



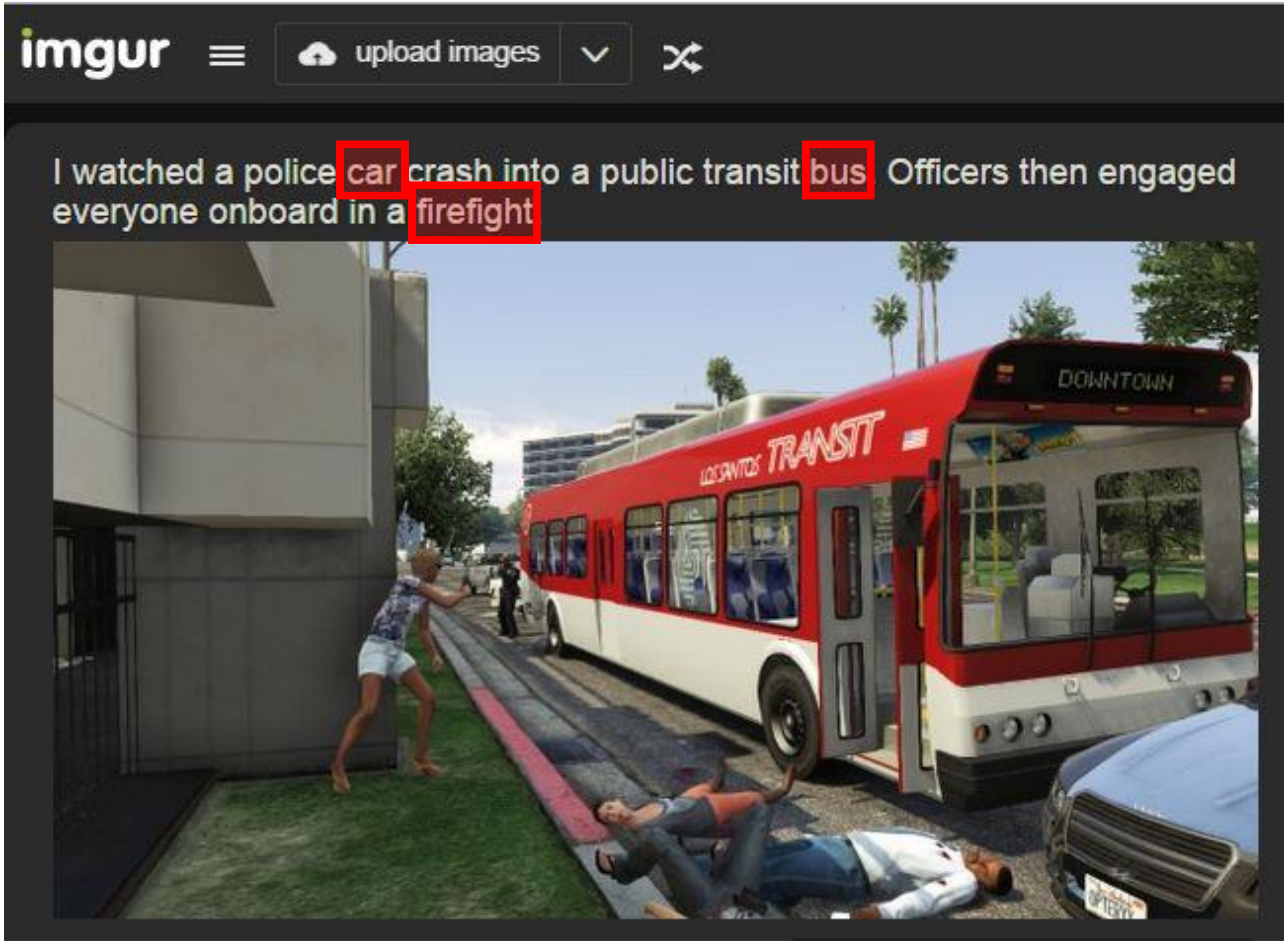
Gordon Roqué @gordonroque · May 27
Today, I am #thankful for my car. I drive a red 2008 #Toyota #Scion xD. Its compact size allows it... instagram.com/p/3NcvlgQUsk/

8:14 PM - 27 May 2015 · Details

Topics:

- Vehicle
- Compact Car

Strong interest in car.



Topics:

- Gaming
- Vehicle
- Battle

Doesn't show interest in car.

Topic Model

- Users' activities are diverse and numerous.
  Dimension reduction is necessary for effective use of the data.
- Topic model is a type of statistical models that find abstract "topics" from a collection of documents, so that each document can be represented by few topics.
- With the topic representation, we can do things like
  - Information Retrieval (find potential car buyer)
  - Classification (or prediction)

- Introduction
- **Dimension Reduction on Texts**
  - TF-IDF (1975)
  - LSI (1990)
  - Aspect Model (1999)
  - LDA (2003)
  - Deep Learning (2006)
- Probabilistic Modeling
  - Graphical Models
  - Multinomial and Dirichlet Distributions
- Latent Dirichlet Allocation
  - Generative Model
  - Inference
  - Applications

TF-IDF (1975)

Salton, Wong and Yong proposed the famous Term Frequency - Inverse Document Frequency formula to represent each document by a fixed-length vector.

$$\text{tfidf}(t, d, D) = \frac{\text{frequcy of term } t \text{ in document } d}{\text{frequency of term } t \text{ in corpus } D}$$

- Dimension reduction is small if vocabulary *V* is large.
- TF-IDF reveals little about statistical structure of the document.

Latent Semantic Indexing (1990)

SVD (Singular Value Decomposition) is the general dimension reduction technique on real-valued vectors.

(Deerwester et. al, 1990) applies SVD to identify the principle components of the tf-idf vectors and use them to represent the documents.

- Deerwester et al. argue that the derived features can capture some aspects of basic linguistic notions such as synonymy.
- LSI is not a generative model.

Aspect Model (1999)

(Hofmann 1999) proposed pLSI (probabilistic LSI) model, aka. aspect model, as an alternative to LSI. pLSI models each word *w* as a sample from a mixture model.

$$P(w \mid d) = \sum_{z \in Z} P(w \mid z) P(z \mid d)$$

The mixture components are multinomial random variables *z* that can be viewed as "topics".

Aspect Model (1999)

The generative model is defined as follows

1. Select a document with probability $P(d),\ d \in D$
2. Pick a latent topic z acc. to topic proportion $P(z \mid d)$
3. Generate a word w acc. to word proportion $P(w \mid z)$

$$P(w \mid d) = \sum_{z \in Z} P(w \mid z) P(z \mid d)$$

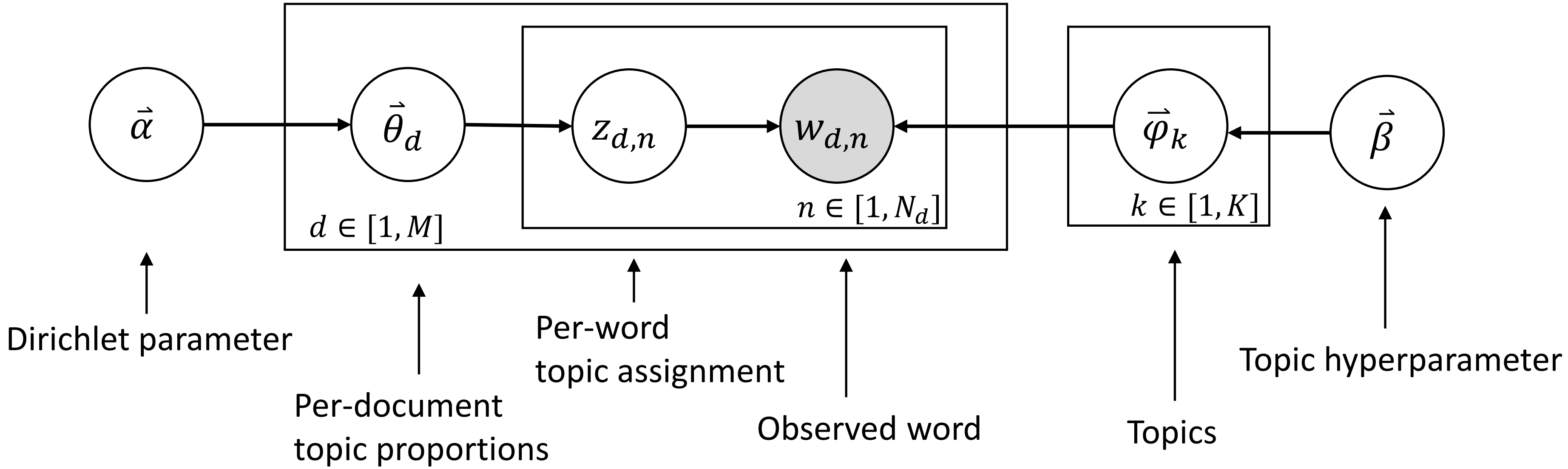
$$P(w, d) = P(w \mid d) P(d)$$

No probabilistic model at the level of documents.

Latent Dirichlet Allocation (2003)

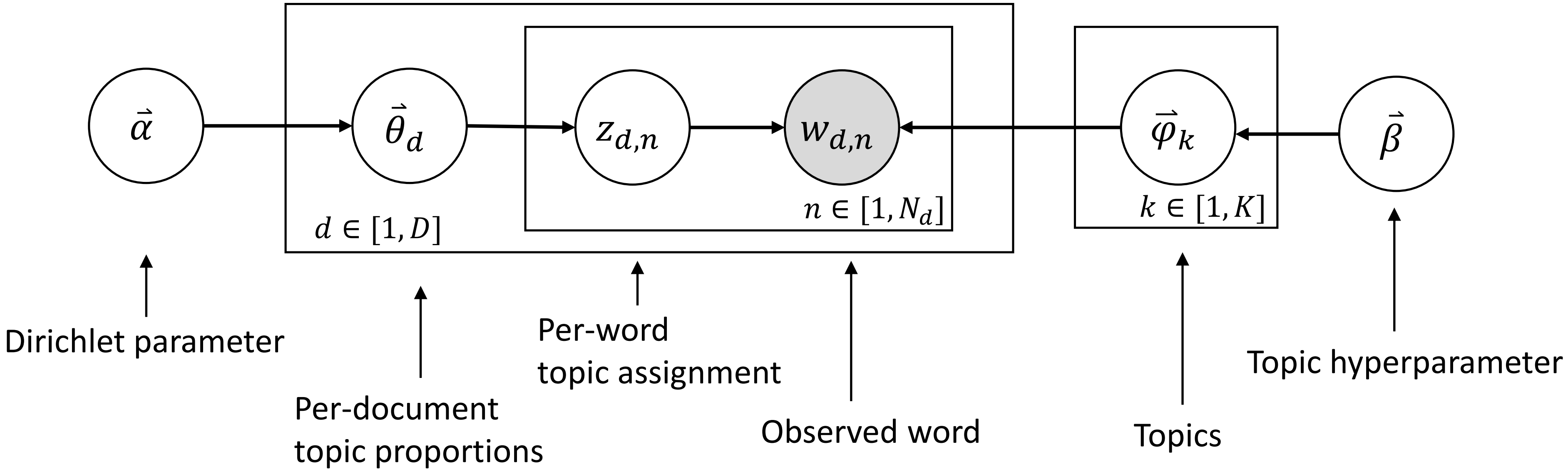
Add Dirichlet priors on aspect model, we get LDA model

$$P\left(\vec{w}_d, \vec{z}_d, \vec{\theta}_d, \vec{\varphi} | \vec{\alpha}, \vec{\beta}\right) = \underbrace{\prod_{n=1}^{N_d} \mathrm{Mult}(w_{d,n} \mid \vec{\varphi}_{z_{d,n}})\, \mathrm{Mult}\left(z_{d,n} \mid \vec{\theta}_d\right)}_{\text{word likelihood}} \cdot \overbrace{\mathrm{Dir}\left(\vec{\theta}_d \mid \vec{\alpha}\right)}^{\text{document prior}} \underbrace{\mathrm{Dir}\left(\vec{\varphi} \mid \vec{\beta}\right)}_{\text{topic prior}}$$



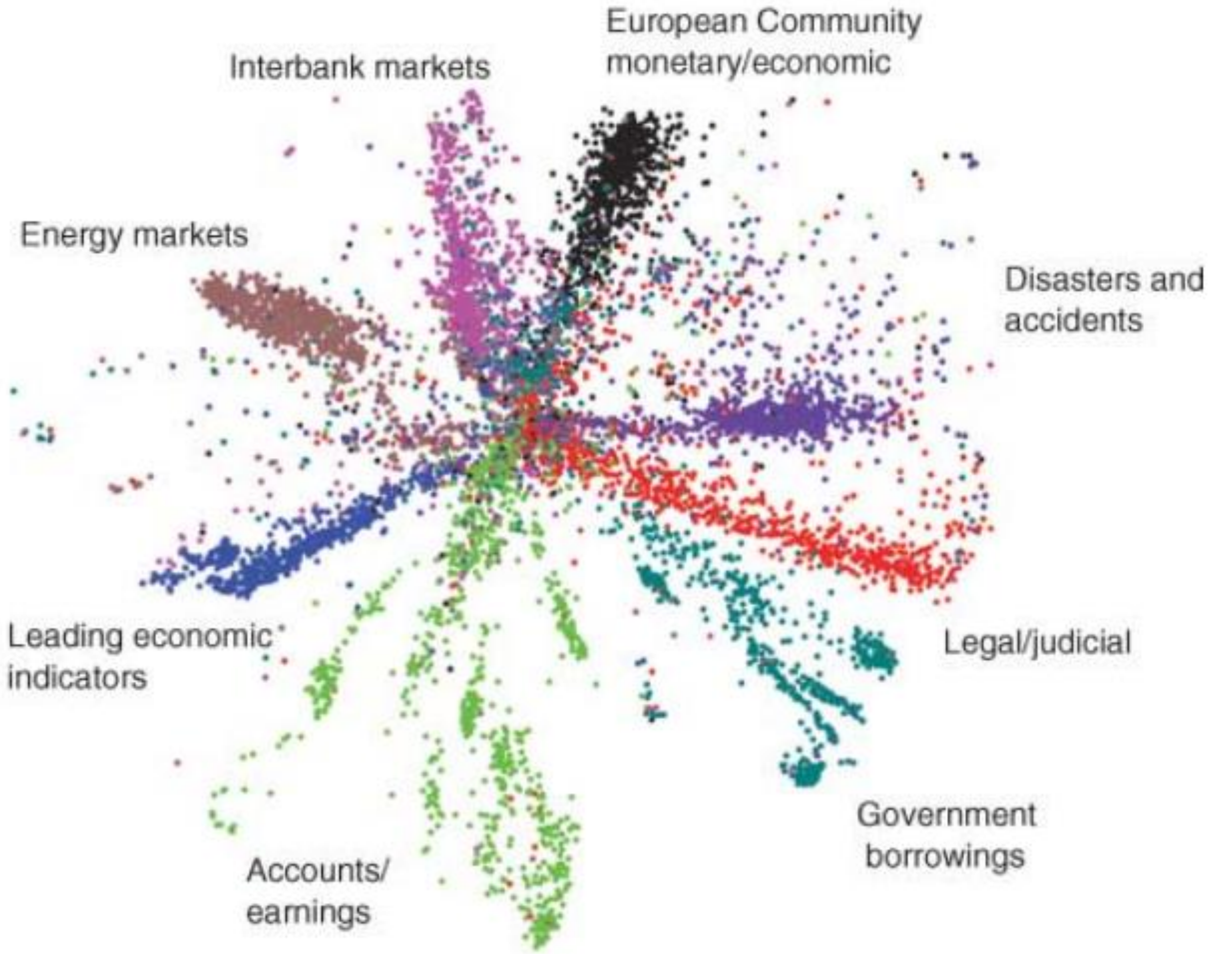
Latent Dirichlet Allocation (Blei et. al. 2003)

1. For each topic $k \in [1, K]$, sample word proportion $\vec{\varphi}_k \sim \mathrm{Dir}(\vec{\beta})$
2. For each document $d \in [1, D]$, sample topic proportion $\vec{\theta}_d \sim \mathrm{Dir}(\vec{\alpha})$
    1. For each word $n \in [1, N_d]$, sample topic index $z_{d,n} \sim \mathrm{Mult}(\vec{\theta}_d)$
    2. For each word $n \in [1, N_d]$, sample term $w_{d,n} \sim \mathrm{Mult}(\vec{\varphi}_{z_{d,n}})$



Deep Learning (2006)

G. E. Hinton and R. R. Salakhutdinov converts high-dimensional data to low-dimensional codes by training a multilayer neural network with a small central layer to reconstruct high-dimensional input vectors.



- Introduction
- Dimension Reduction on Texts
  - TF-IDF (1975)
  - LSI (1990)
  - Aspect Model (1999)
  - LDA (2003)
  - Deep Learning (2006)
- **Probabilistic Modeling**
  - Graphical Models
  - Multinomial and Dirichlet Distributions
- Latent Dirichlet Allocation
  - Generative Model
  - Inference
  - Applications

Probabilistic Modeling

1. Treat data as observations that arise from a generative probabilistic process that includes hidden variables.
   - For documents, the hidden variables reflect the thematic structure of a document.
2. Infer the hidden structure (topics) using posterior inference.
   - MCMC (Markov Chain Monte Carlo)

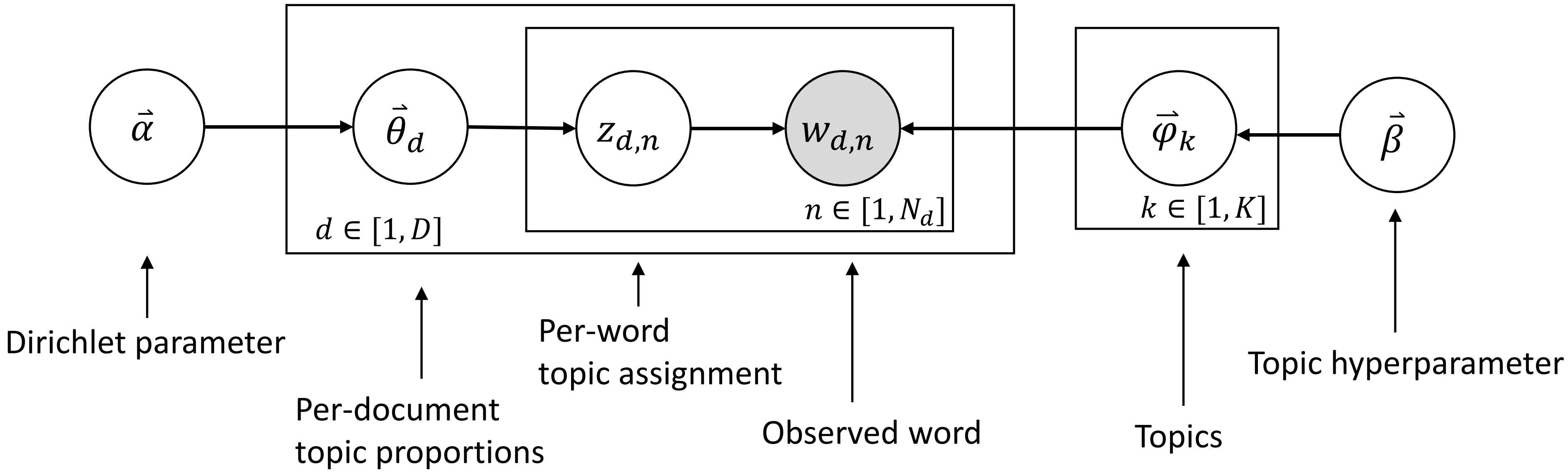
$$P(\theta \mid X) = \frac{P(X \mid \theta)P(\theta)}{P(X)}$$

$$\text{posterior} = \frac{\text{likelihood} * \text{prior}}{\text{evidence}}$$

3. Situate new data into the estimated model.

Graphical Models

- Nodes are random variables
- Edges denote possible dependence.
- Observed variables are shaded.
- Replicate structures are boxed.



Multinomial and Dirichlet Distributions

- Multinomial distribution conditioned on word proportion $\vec{p}$

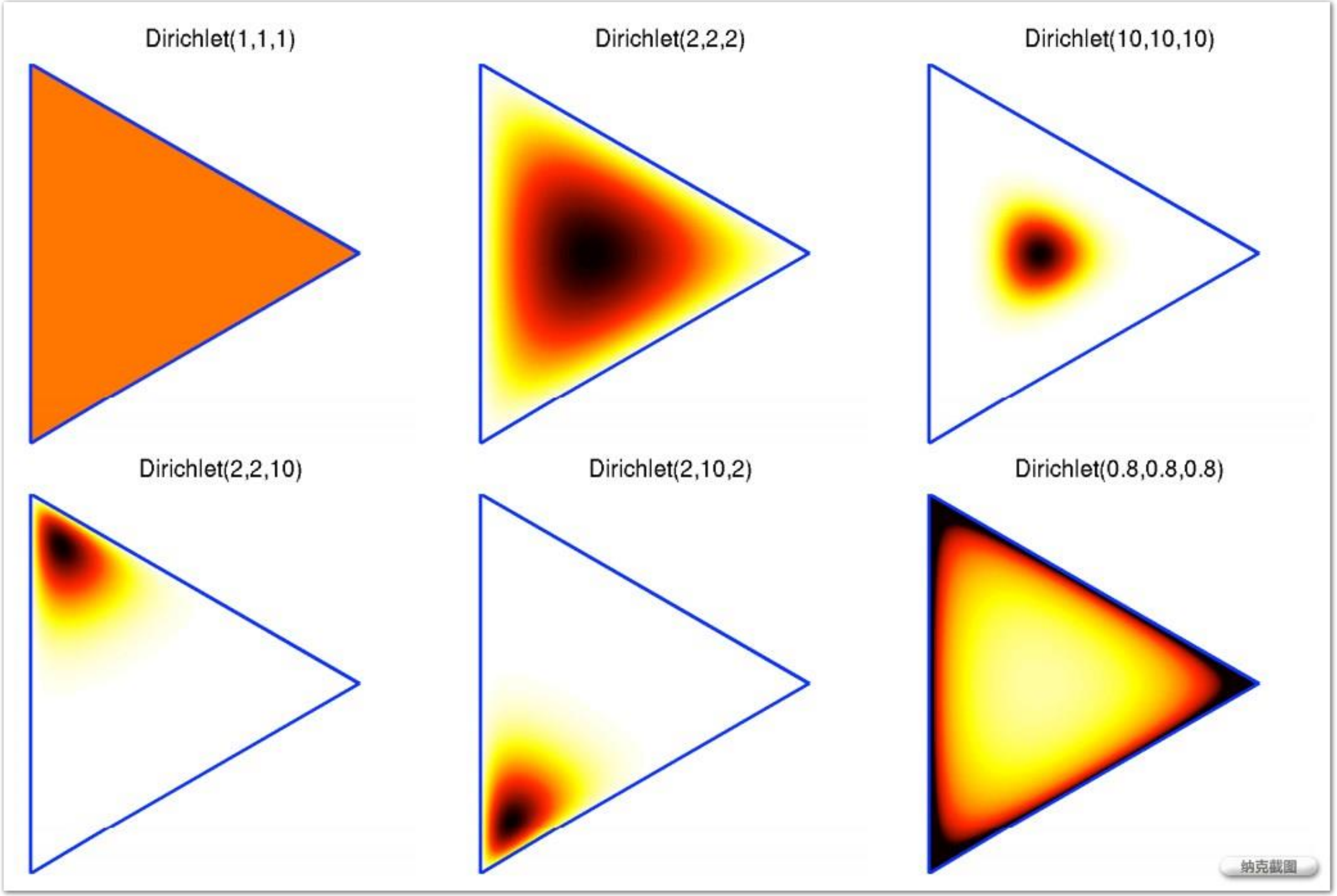
$$P(\vec{w} \mid \vec{p}) \;=\; \mathrm{Mult}(\vec{n} \mid \vec{p}) \;=\; \frac{\Gamma\left(\sum_t n^{(t)}+1\right)}{\prod_t \Gamma(n^{(t)}+1)} \prod_{t=1}^{V} p_t^{n^{(t)}}$$

- Dirichlet distribution conditioned on pseudo-count $\vec{\alpha}$

$$P(\vec{p} \mid \vec{\alpha}) \;=\; \mathrm{Dir}(\vec{p} \mid \vec{\alpha}) \;=\; \frac{1}{\Delta(\vec{\alpha})} \prod_{t=1}^{V} p_t^{\alpha_t - 1}$$

$$\Delta(\vec{\alpha}) \;=\; \frac{\prod_t \Gamma(\alpha_t)}{\Gamma\left(\sum_t \alpha_t\right)}$$

Dirichlet Pseudo-Count when K=3



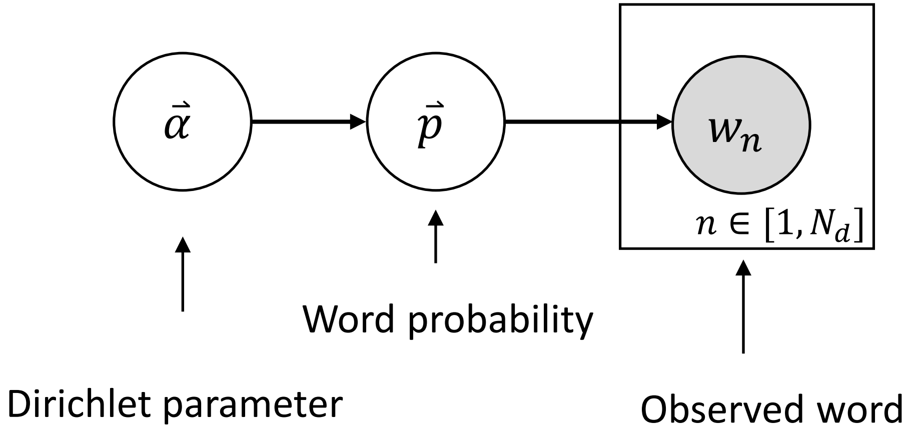
Multinomial and Dirichlet Distributions

- Thanks to conjugacy, the Dirichlet posterior given multinomial observation $\vec{w}$ and pseudo-counts $\vec{\alpha}$ is

$$P(\vec{p} \mid \vec{w}, \vec{\alpha}) = \frac{\mathrm{Mult}(\vec{w} \mid \vec{p})\ \mathrm{Dir}(\vec{p} \mid \alpha)}{P(\vec{w} \mid \vec{\alpha})} = \mathrm{Dir}(\vec{p} \mid \vec{\alpha} + \vec{n})$$

- Likelihood$(\vec{p}) \sim \mathrm{Mult}(\vec{w} \mid \vec{p})$
- Prior$(\vec{p}) \sim \mathrm{Dir}(\vec{p} \mid \vec{\alpha})$

- Having new observation is equivalent with adjusting pseudo-count.

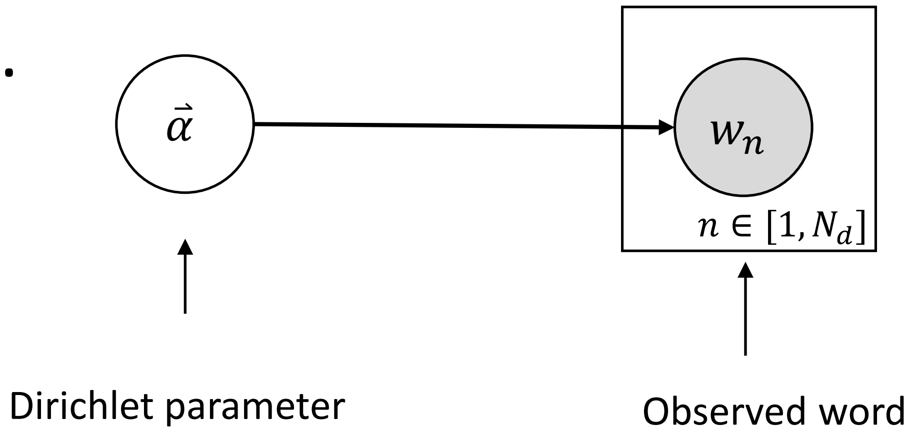


Multinomial and Dirichlet Distributions

- Integrate out *p* to get

$$
\begin{aligned}
P(\vec{w} \mid \vec{\alpha}) &= \int_{\vec{p}} \mathrm{Mult}(\vec{w} \mid \vec{p})\ \mathrm{Dir}(\vec{p} \mid \alpha)\ d\vec{p} \\
&= \frac{\Delta(\vec{n} + \vec{\alpha})}{\Delta(\vec{\alpha})}
\end{aligned}
$$

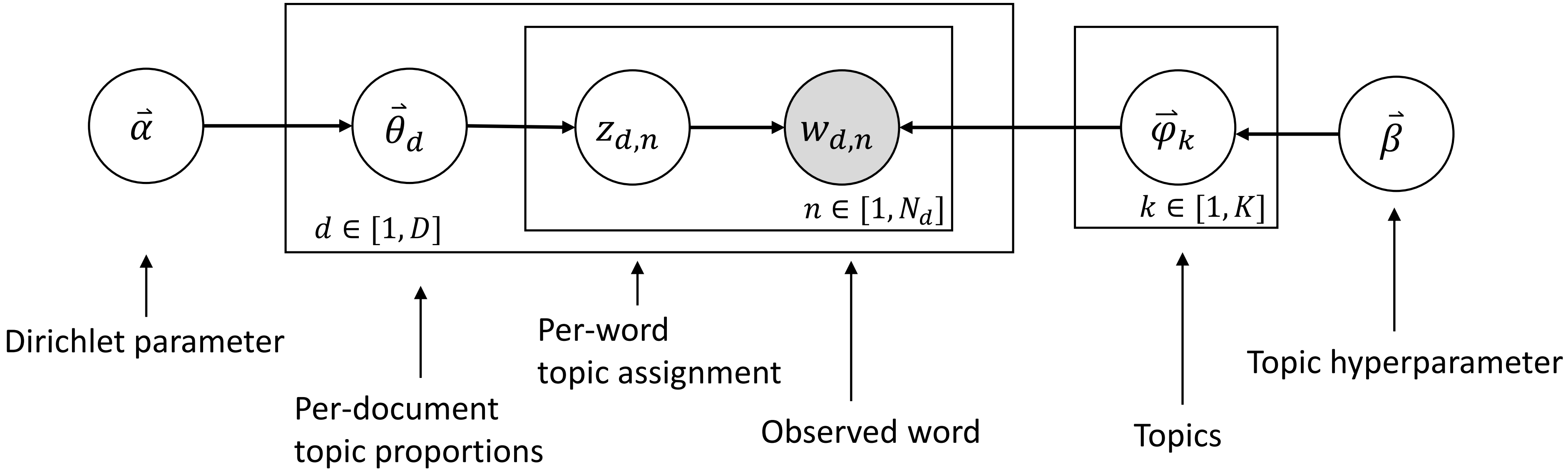
- We'll use this repeatedly in later derivation.



- Introduction
- Dimension Reduction on Texts
  - TF-IDF (1975)
  - LSI (1990)
  - Aspect Model (1999)
  - LDA (2003)
  - Deep Learning (2006)
- Probabilistic Modeling
  - Graphical Models
  - Multinomial and Dirichlet Distributions
- **Latent Dirichlet Allocation**
  - Generative Model
  - Inference
  - Applications

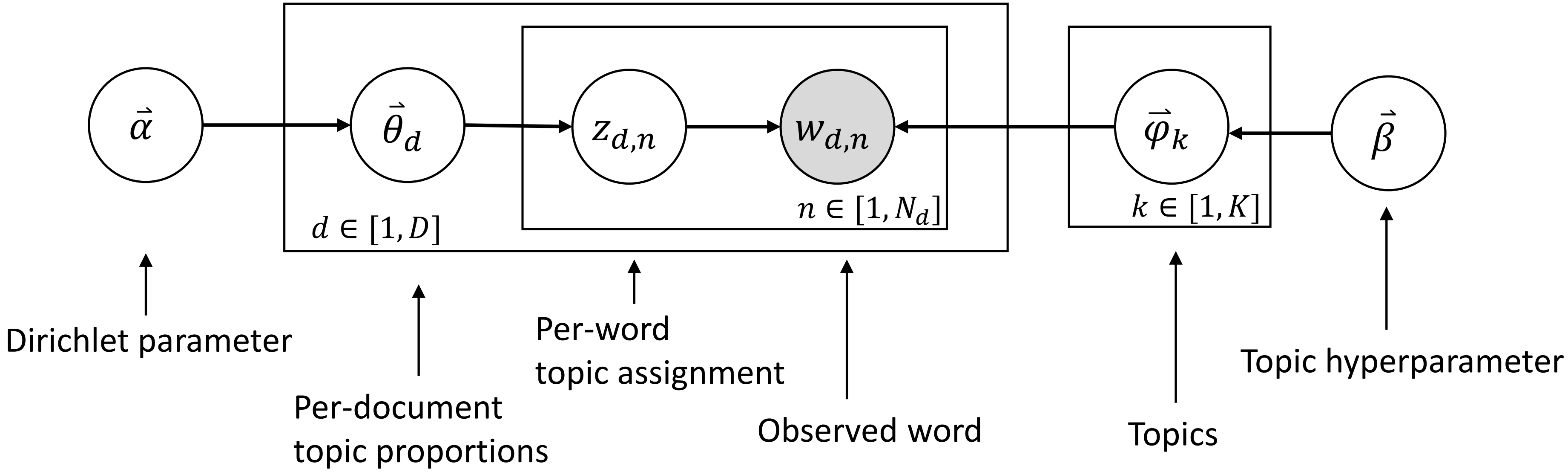
Generative Model

1. For each topic $k \in [1, K]$, sample word proportion $\vec{\varphi}_k \sim \mathrm{Dir}(\vec{\beta})$
2. For each document $d \in [1, D]$, sample topic proportion $\vec{\theta}_d \sim \mathrm{Dir}(\vec{\alpha})$
   1. For each word $n \in [1, N_d]$, sample topic index $z_{d,n} \sim \mathrm{Mult}(\vec{\theta}_d)$
   2. For each word $n \in [1, N_d]$, sample term $w_{d,n} \sim \mathrm{Mult}(\vec{\varphi}_{z_{d,n}})$



Generative Model

$$P\left(\vec{w}_d, \vec{z}_d, \vec{\theta}_d, \vec{\varphi} \mid \vec{\alpha}, \vec{\beta}\right) = \underbrace{\prod_{n=1}^{N_d} \mathrm{Mult}(w_{d,n} \mid \vec{\varphi}_{z_{d,n}})\,\mathrm{Mult}\left(z_{d,n} \mid \vec{\theta}_d\right)}_{\text{word likelihood}} \cdot \overbrace{\mathrm{Dir}\left(\vec{\theta}_d \mid \vec{\alpha}\right)}^{\text{document prior}}\underbrace{\mathrm{Dir}\left(\vec{\varphi} \mid \vec{\beta}\right)}_{\text{topic prior}}$$



Inference

- Given observations *W*, we want to infer the hidden structure (topics) from posterior probability.
- Posterior probability is computationally intractable. Approximate inference is used.
  - MCMC (Markov Chain Monte Carlo): Gibbs sampler, collapse Gibbs sampler.
  - Variational methods: replace sampling with optimization.
  - “Distributed Algorithms for Topic Models” by D. Newman, A. Asuncion et. al., 2009.

Inference - Strategy

- Given document $\vec{w}$, sample topic assignment $\vec{z}$ from posterior

$$P\left(\vec{z} \mid \vec{w}, \vec{\alpha}, \vec{\beta}\right) = \frac{P\left(\vec{w} \mid \vec{z}, \vec{\beta}\right) P(\vec{z} \mid \vec{\alpha})}{P\left(\vec{w} \mid \vec{\alpha}, \vec{\beta}\right)}$$

- Estimate topic proportion and word proportion by Dirichlet posterior

$$P\left(\vec{\theta}_d \mid \vec{z}_d, \vec{\alpha}\right) = \mathrm{Dir}\left(\vec{\theta}_d \mid \vec{\alpha} + \vec{n}_d\right)$$

$$P\left(\vec{\varphi}_k \mid \vec{z}, \vec{w}, \vec{\beta}\right) = \mathrm{Dir}\left(\vec{\varphi}_k \mid \vec{\beta} + \vec{n}_k\right)$$

$$\mathbf{E}(\theta_{d,k}) = \frac{n_d^{(k)} + \alpha_k}{\sum_{k'=1}^{K} n_d^{(k')} + \alpha_{k'}}$$

$$\mathbf{E}(\varphi_{k,t}) = \frac{n_k^{(t)} + \beta_t}{\sum_{t'=1}^{V} n_k^{(t')} + \beta_{t'}}$$

Inference – Likelihood

- Given document $\vec{w}$, sample topic assignment $\vec{z}$ from posterior

$$P\left(\vec{z} \mid \vec{w}, \vec{\alpha}, \vec{\beta}\right) = \frac{P\left(\vec{w} \mid \vec{z}, \vec{\beta}\right) P(\vec{z} \mid \vec{\alpha})}{P\left(\vec{w} \mid \vec{\alpha}, \vec{\beta}\right)}$$

- Likelihood

$$\begin{aligned}
P\left(\vec{w} \mid \vec{z}, \vec{\beta}\right) &= \int P(\vec{w} \mid \vec{z}, \vec{\varphi}) P\left(\vec{\varphi} \mid \vec{\beta}\right) d\vec{\varphi} \\
&= \int \prod_{k=1}^{K} \mathrm{Mult}(\vec{n}_k \mid \vec{\varphi}_k)\, \mathrm{Dir}\left(\vec{\varphi}_k \mid \vec{\beta}\right) d\vec{\varphi} \\
&= \prod_{k=1}^{K} \frac{\Delta\left(\vec{n}_k + \vec{\beta}\right)}{\Delta\left(\vec{\beta}\right)}
\end{aligned}$$

Inference – Prior

- Given document $\vec{w}$, sample topic assignment $\vec{z}$ from posterior

$$P\left(\vec{z} \mid \vec{w}, \vec{\alpha}, \vec{\beta}\right) = \frac{P\left(\vec{w} \mid \vec{z}, \vec{\beta}\right) P(\vec{z} \mid \vec{\alpha})}{P\left(\vec{w} \mid \vec{\alpha}, \vec{\beta}\right)}$$

- Prior

$$\begin{aligned} P(\vec{z} \mid \vec{\alpha}) &= \int P\left(\vec{z} \mid \vec{\theta}\right) P\left(\vec{\theta} \mid \vec{\alpha}\right) d\vec{\theta} \\ &= \int \prod_{d=1}^{D} \mathrm{Mult}\left(\vec{n}_d \mid \vec{\theta}_d\right) \mathrm{Dir}\left(\vec{\theta}_d \mid \vec{\alpha}\right) d\vec{\theta} \\ &= \prod_{d=1}^{D} \frac{\Delta(\vec{n}_d + \vec{\alpha})}{\Delta(\vec{\alpha})} \end{aligned}$$

Inference – Collapsed Conditional

- Given document $\vec{w}$, sample topic assignment $\vec{z}$ from posterior

$$P\left(\vec{z} \mid \vec{w}, \vec{\alpha}, \vec{\beta}\right) = \frac{P\left(\vec{w} \mid \vec{z}, \vec{\beta}\right) P(\vec{z} \mid \vec{\alpha})}{P\left(\vec{w} \mid \vec{\alpha}, \vec{\beta}\right)}$$

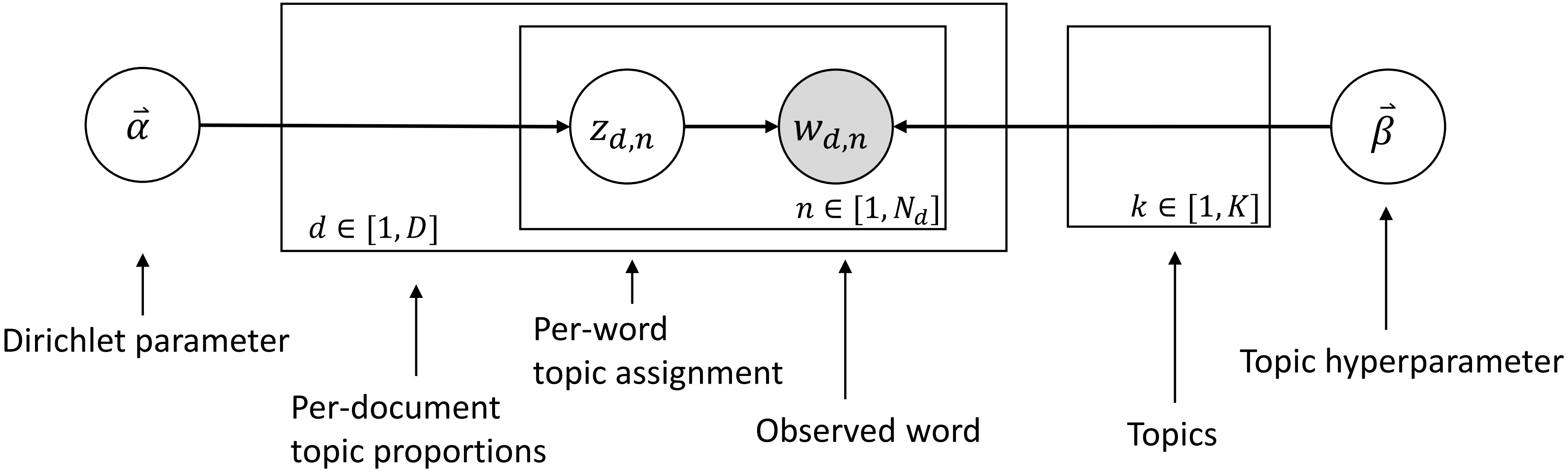
- For $\vec{w} = \{w_i = t, \vec{w}_{\neg i}\}$ and $\vec{z} = \{z_i = k, \vec{z}_{\neg i}\}$

$$P\left(z_i = k \mid \vec{z}_{\neg i}, \vec{w}, \vec{\alpha}, \vec{\beta}\right) = \frac{P\left(\vec{w} \mid \vec{z}, \vec{\beta}\right) P(\vec{z} \mid \vec{\alpha})}{P\left(\vec{w}, \vec{z}_{\neg i} \mid \vec{\alpha}, \vec{\beta}\right)} = \frac{P\left(\vec{w} \mid \vec{z}, \vec{\beta}\right)}{P\left(\vec{w}_{\neg i} \mid \vec{z}_{\neg i}, \vec{\beta}\right) P(w_i)} \frac{P(\vec{z} \mid \vec{\alpha})}{P(\vec{z}_{\neg i} \mid \vec{\alpha})}$$

$$\propto \frac{\Delta\left(\vec{n}_z + \vec{\beta}\right)}{\Delta\left(\vec{n}_{z,\neg i} + \vec{\beta}\right)} \frac{\Delta(\vec{n}_d + \vec{\alpha})}{\Delta(\vec{n}_{m,\neg i} + \vec{\alpha})}$$

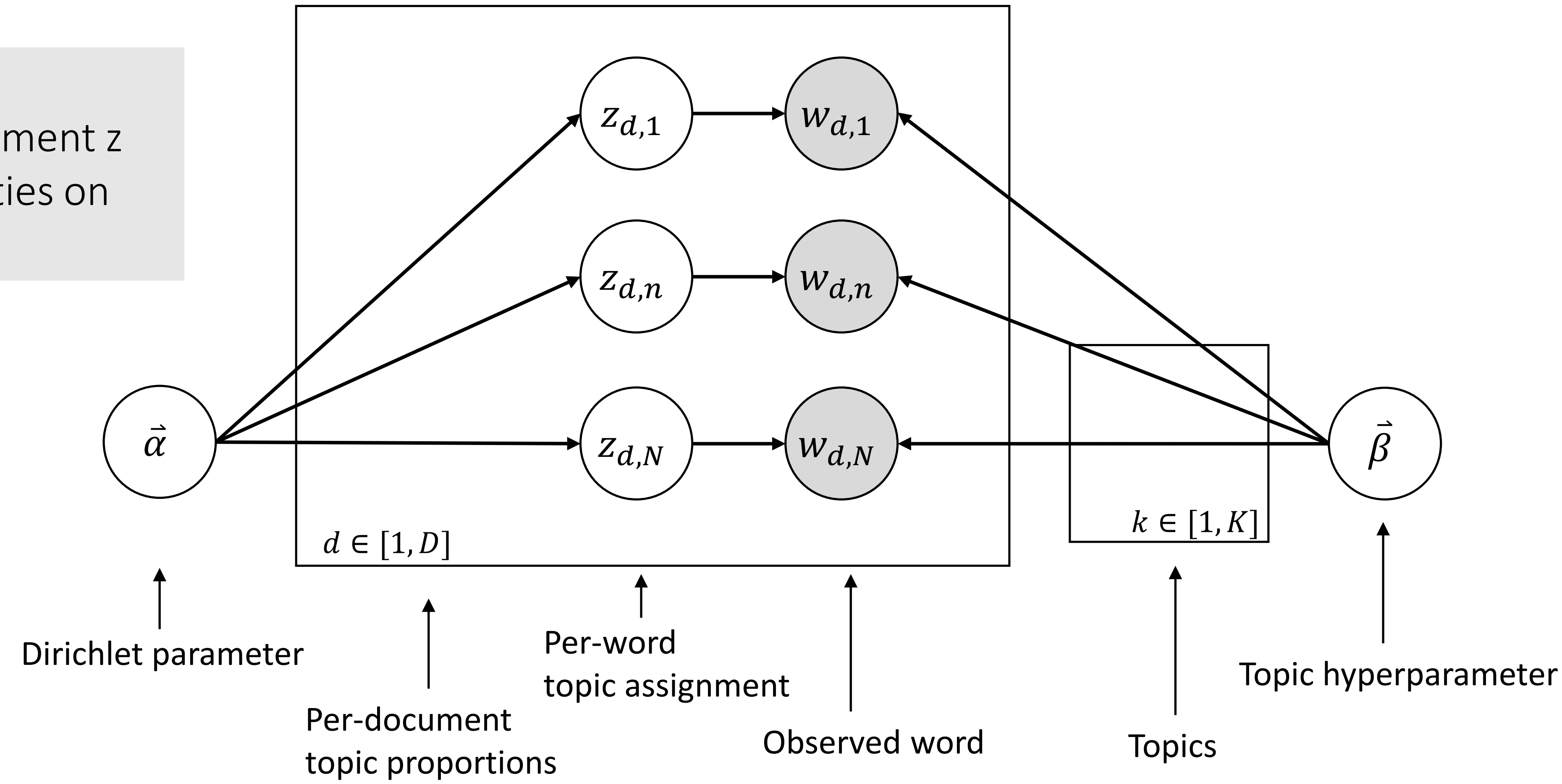
$$\propto \frac{n_{k,\neg i}^{(t)} + \beta_t}{\sum_{t'=1}^{V} n_{k,\neg i}^{(t')} + \beta_t'} \left(n_{d,\neg i}^{(k)} + \alpha_k\right)$$

Inference – Collapsed Conditional



Inference – Gibbs Sampling

Stage 1 Initialization:
Initialize topic assignment z with equal probabilities on topics.

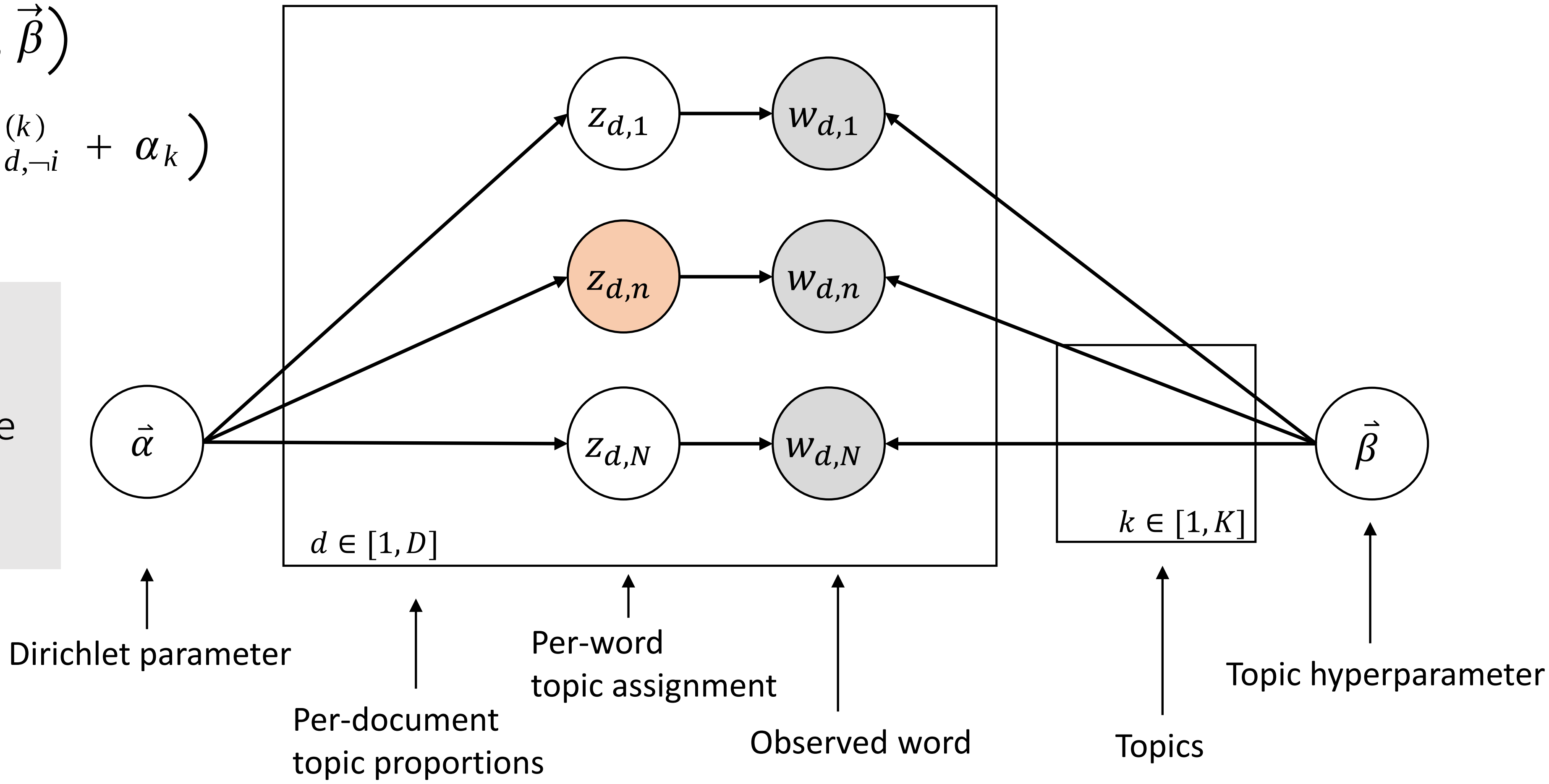


Inference – Gibbs Sampling

$$P\left(z_i = k \,|\, \vec{z}_{\neg i}, \vec{w}, \vec{\alpha}, \vec{\beta}\right)$$

$$\propto \frac{n_{k,\neg i}^{(t)} + \beta_t}{\sum_{t'=1}^{V} n_{k,\neg i}^{(t')} + \beta_t'} \left(n_{d,\neg i}^{(k)} + \alpha_k\right)$$

Stage 2 Burn-In Period:
Randomly select topic assignment z. Update the assignment acc. to the collapsed conditional.

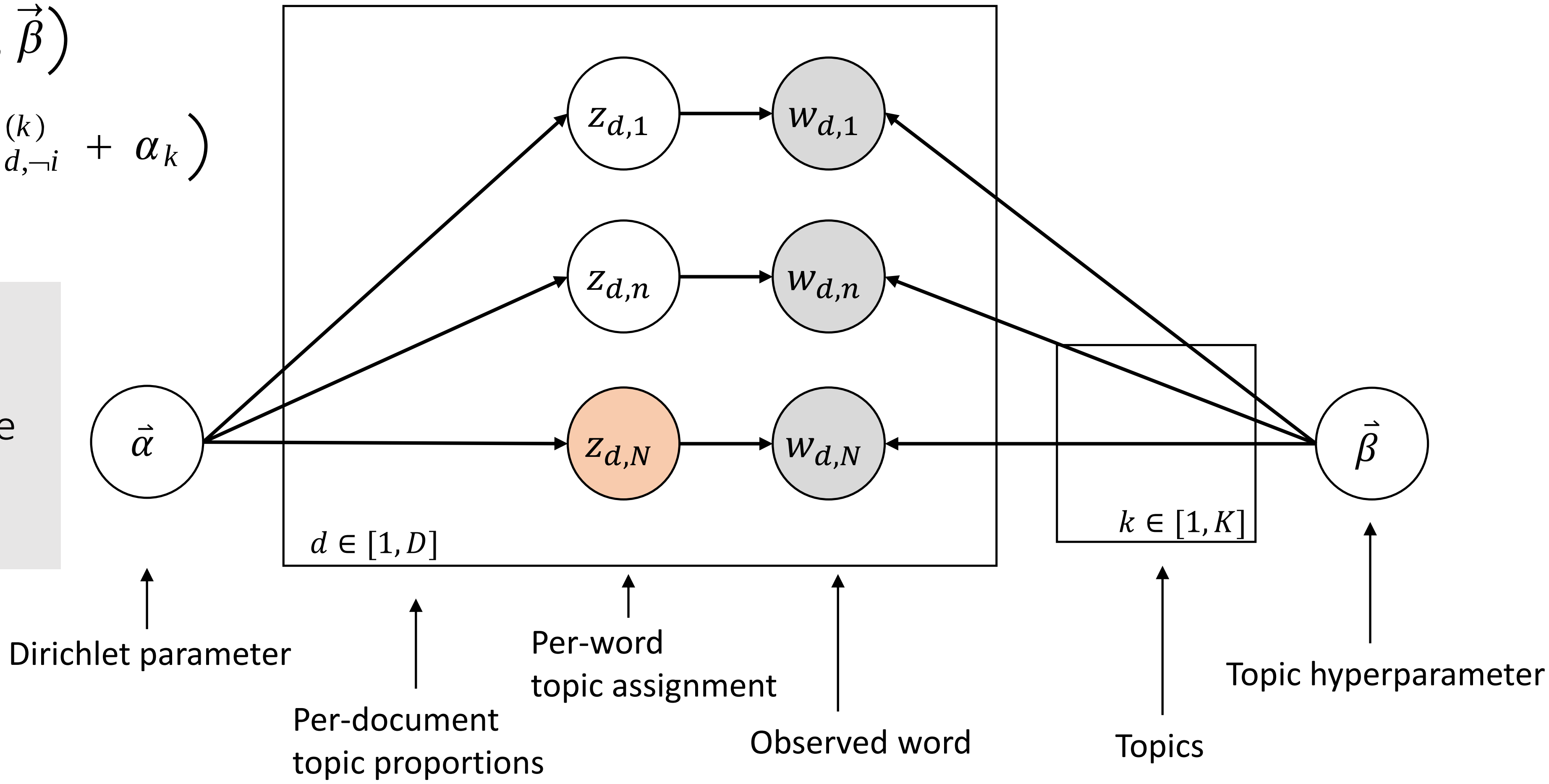


Inference – Gibbs Sampling

$$P\left(z_i = k \mid \vec{z}_{\neg i}, \vec{w}, \vec{\alpha}, \vec{\beta}\right)$$

$$\propto \frac{n_{k,\neg i}^{(t)} + \beta_t}{\sum_{t'=1}^{V} n_{k,\neg i}^{(t')} + \beta'_t} \left(n_{d,\neg i}^{(k)} + \alpha_k\right)$$

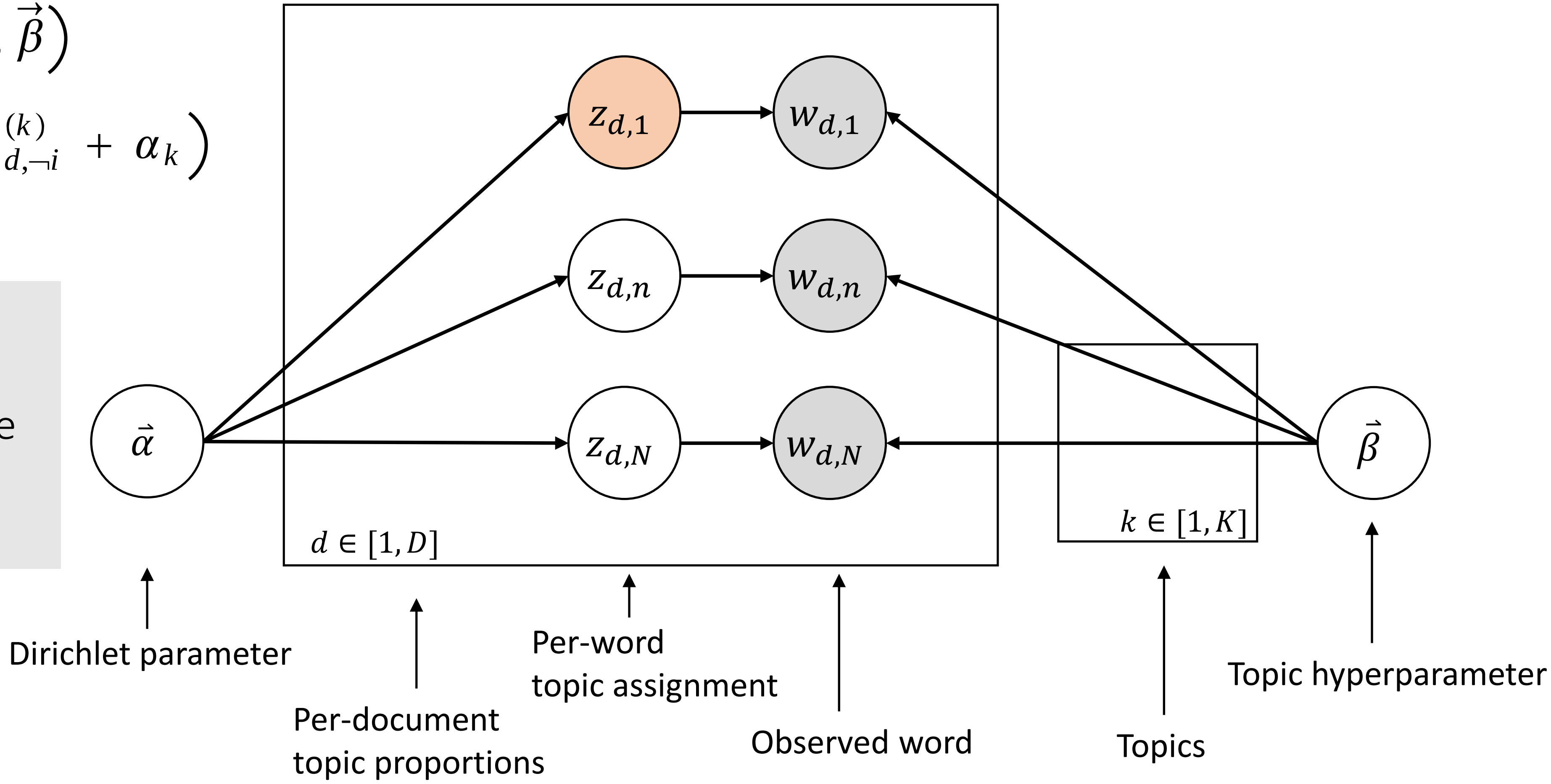
Stage 2 Burn-In Period: Randomly select topic assignment z. Update the assignment acc. to the collapsed conditional.



Inference – Gibbs Sampling

$$P\left(z_i = k \mid \vec{z}_{\neg i}, \vec{w}, \vec{\alpha}, \vec{\beta}\right) \propto \frac{n_{k,\neg i}^{(t)} + \beta_t}{\sum_{t'=1}^{V} n_{k,\neg i}^{(t')} + \beta'_t} \left(n_{d,\neg i}^{(k)} + \alpha_k\right)$$

Stage 2 Burn-In Period:
Randomly select topic assignment z. Update the assignment acc. to the collapsed conditional.



Inference – Gibbs Sampling

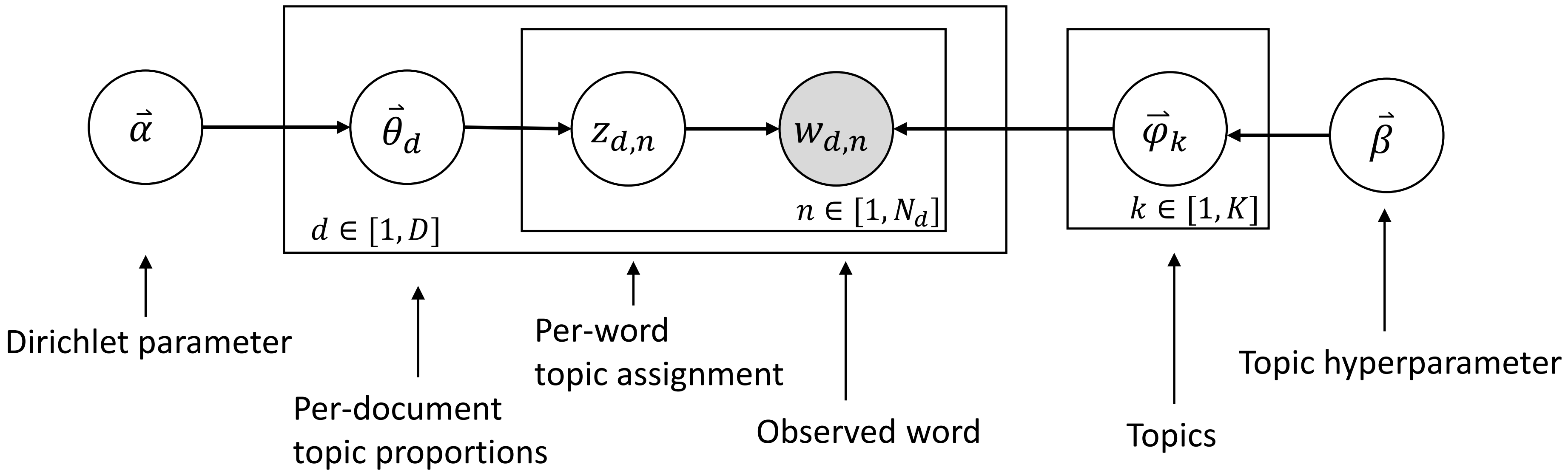
$$P\left(\vec{\theta}_d \mid \vec{z}_d, \vec{\alpha}\right) = \mathrm{Dir}\left(\vec{\theta}_d \mid \vec{\alpha} + \vec{n}_d\right)$$

$$P\left(\vec{\varphi}_k \mid \vec{z}, \vec{w}, \vec{\beta}\right) = \mathrm{Dir}\left(\vec{\varphi}_k \mid \vec{\beta} + \vec{n}_k\right)$$

$$\mathbf{E}(\theta_{d,k}) = \frac{n_d^{(k)} + \alpha_k}{\sum_{k'=1}^{K} n_d^{(k')} + \alpha_{k'}}$$

$$\mathbf{E}(\varphi_{k,t}) = \frac{n_k^{(t)} + \beta_t}{\sum_{t'=1}^{V} n_k^{(t')} + \beta_{t'}}$$

Stage 3 Terminate:
When stop criteria is met, read out word proportion and topic proportion from topic assignment counts.



Inference – Gibbs Sampler Initialization

$$\text{Algorithm}\left\{\vec{\varphi}, \vec{\theta}, \vec{z}\right\} \quad = \text{LDA\_Gibbs}\left(\{\vec{w}\}, \vec{\alpha}, \vec{\beta}, K\right)$$

//initialize counts and topic assignment $\vec{z}$

sample topic indices $\vec{z} \sim \text{Mult}(1/K)$ for every word in $\{\vec{w}\}$

set document-topic count $n_d^{(k)} = \|\{z_{d,n} = k \mid n \in [1, N_d]\}\|$

set topic-term count $n_k^{(t)} = \|\{w_{d,n} = t \mid z_{d,n} = k\}\|$

Inference – Gibbs Sampler Burn-In Period

while not finished **do**
  for all documents $d \in [1, D]$ **do**
    //for the current assignment of topic k to term t for word $w_{d,n}$
    decrease counts: $n_d^{(k)} \;-\!=\; 1,\;\; n_k^{(t)} \;-\!=\; 1$
    //multinomial sampling for each component i
    sample topic index $k' \sim P\left(z_i \mid \vec{z}_{\neg i}, \vec{w}, \vec{\alpha}, \vec{\beta}\right)$
    //for the new assignment of $z_{d,n}$ to the term t for word $w_{d,n}$
    increament counts: $n_d^{(k')} \;+\!=\; 1,\;\; n_{k'}^{(t)} \;+\!=\; 1$
  end for

  if converged or L sampling iterations since last read out **then**
    read out parameter estimates $\mathbf{E}(\theta_{d,k})$ and $\mathbf{E}(\varphi_{k,t})$.
  end if
end while

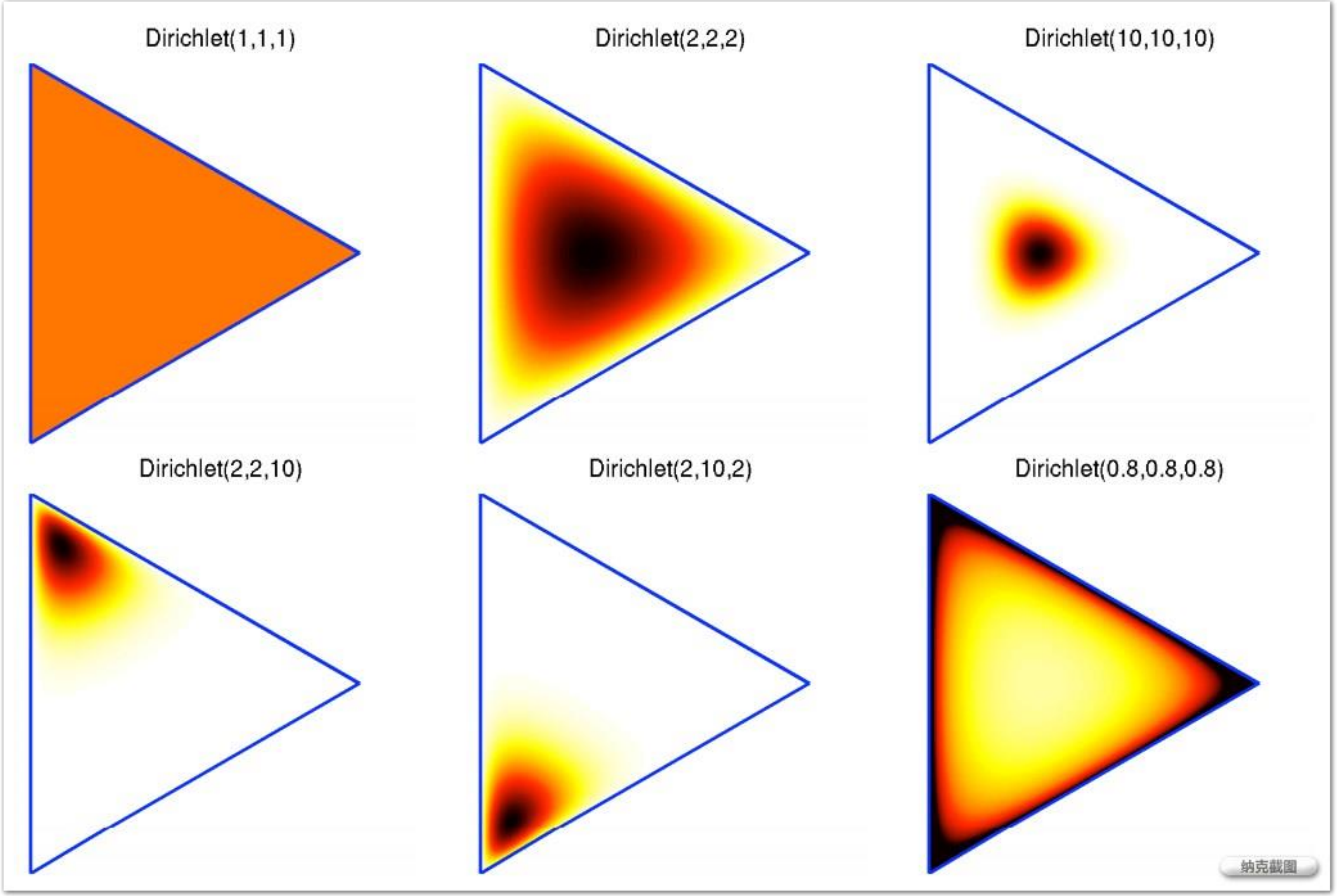
Inference - Dirichlet Parameters

- Dirichlet parameters describe our belief in the outcome.
- Dirichlet parameters less than one penalize dense topic/word proportions.

- Suggested value [Griffiths&Steyvers 2004]

$$\begin{aligned} \alpha &= 50/K \\ \beta &= 0.01 \end{aligned}$$

Dirichlet Pseudo-Count when K=3



Applicatons

- Given new document $\widetilde{w}$, Gibbs sampling update:

$$P\left(\tilde{z}_i = k \mid \tilde{w}_i = t, \tilde{z}_{\neg i}, \tilde{w}_{\neg i}, \vec{\varphi}, \vec{\theta}\right) \propto \varphi_{k,t} * \left(n_{d',\neg i}^{(k)} + \alpha_k\right)$$

$$\mathbf{E}(\theta_{d',k}) = \frac{n_{d'}^{(k)} + \alpha_k}{\sum_{k'=1}^{K} n_{d'}^{\left(k'\right)} + \alpha_{k'}}$$

- Thus we obtain the topic proportion for documents and queries. Information retrieval and classification can be carried out on top of this representation.

References

- Blei, David M., Andrew Y. Ng, and Michael I. Jordan. "Latent dirichlet allocation." *the Journal of machine Learning research* 3 (2003)
- Heinrich, Gregor. ”Parameter estimation for text analysis.” Technical report, 2005.
- Hofmann, Thomas. "Probabilistic latent semantic indexing." *Proceedings of the 22nd annual international ACM SIGIR*, 1999.
- http://videolectures.net/mlss09uk_blei_tm by David Blei