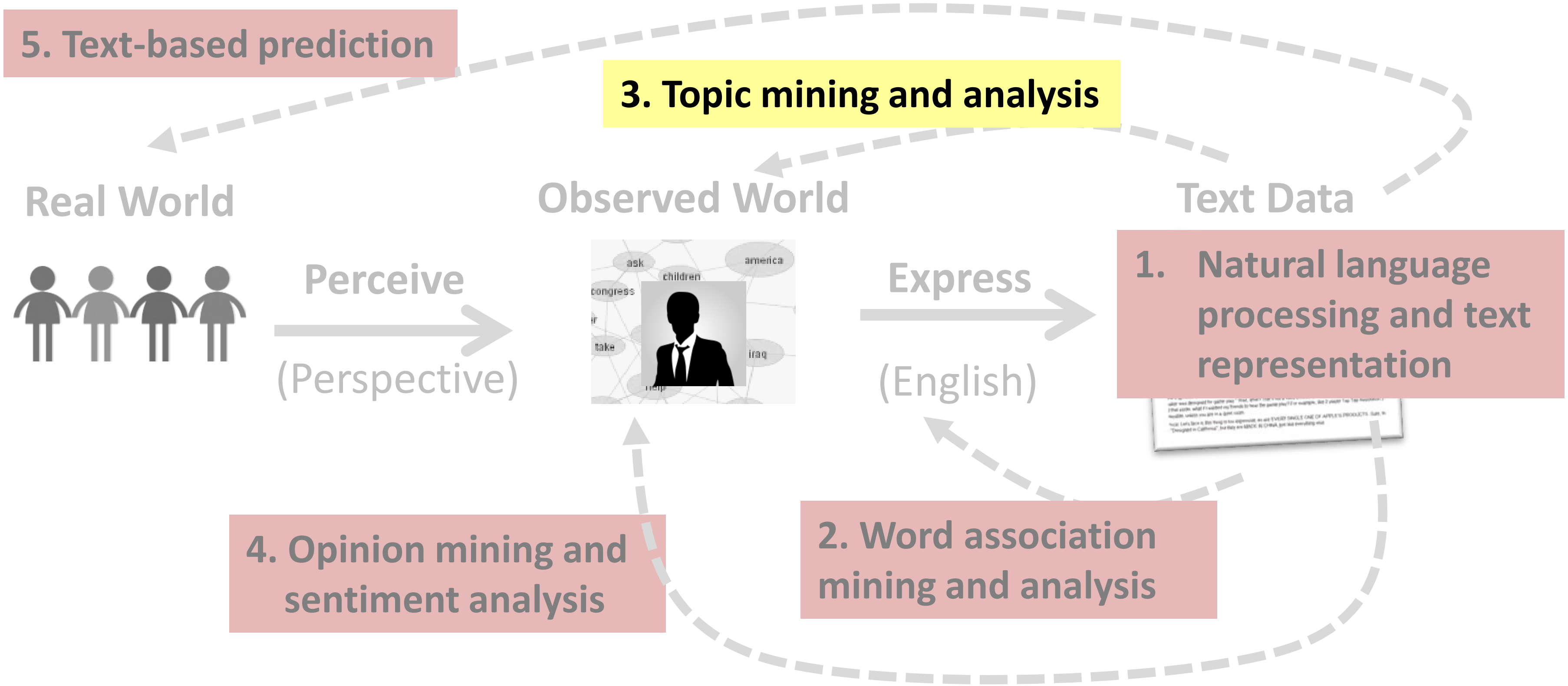
Text Clustering: Generative Probabilistic Models

Part 3

ChengXiang “Cheng” Zhai
Department of Computer Science
University of Illinois at Urbana-Champaign

Text Clustering: Generative Probabilistic Models (Part 3)



How Can We Compute the ML Estimate?

- Data: a collection of documents $C=\{d_1, \ldots, d_N\}$
- Model: mixture of k unigram LMs: $\Lambda=(\{\theta_i\}; \{p(\theta_i)\})$, $i \in [1,k]$
  - To generate a document, first **choose a θ_i** according to $p(\theta_i)$ and then generate **all** words in the document using $p(w|\theta_i)$
- Likelihood:

$$p(d|\Lambda) = \sum_{i=1}^{k} [p(\theta_i) \prod_{w \in V} p(w|\theta_i)^{c(w,d)}]$$

$$p(C|\Lambda) = \prod_{j=1}^{N} p(d_j|\Lambda)$$

- Maximum Likelihood estimate

$$\Lambda^* = \arg\max_{\Lambda} p(C|\Lambda)$$

EM Algorithm for Document Clustering

- Initialization: Randomly set **Λ=({θ_i}; {p(θ_i)}), i$\in$[1,k]**
- **Repeat until likelihood p(C|Λ) converges**
  - **E-Step: Infer which distribution has been used to generate document d: hidden variable $Z_d \in [1, k]$**

$\propto$：正比符号

$$p^{(n)}(Z_d = i \mid d) \propto p^{(n)}(\theta_i) \prod_{w \in V} p^{(n)}(w \mid \theta_i)^{c(w,d)} \qquad \sum_{i=1}^{k} p^{(n)}(Z_d = i \mid d) = 1$$

隐藏变量

  - **M-Step: Re-estimation of all parameters**

$$p^{(n+1)}(\theta_i) \propto \sum_{j=1}^{N} p^{(n)}(Z_{d_j} = i \mid d_j) \qquad \sum_{i=1}^{k} p^{(n+1)}(\theta_i) = 1$$

$$p^{(n+1)}(w \mid \theta_i) \propto \sum_{j=1}^{N} c(w, d_j) p^{(n)}(Z_{d_j} = 1 \mid d_j) \qquad \sum_{w \in V} p^{(n+1)}(w \mid \theta_i) = 1, \quad \forall i \in [1, k]$$

An Example of 2 Clusters

Random Initialization

$p(\theta_1) = p(\theta_2) = 0.5$

	$p(w\|\theta_1)$	$p(w\|\theta_2)$
text	0.5	0.1
mining	0.2	0.1
medical	0.2	0.75
health	0.1	0.05

E-step **Document d**

Hidden variables:

$Z_d \in \{1, 2\}$

	c(w,d)
text	2
mining	2
medical	0
health	0

$$p(Z_d = 1 | d) = \frac{p(\theta_1)p(\text{"text"}|\theta_1)^2 p(\text{"mining"}|\theta_1)^2}{p(\theta_1)p(\text{"text"}|\theta_1)^2 p(\text{"mining"}|\theta_1)^2 + p(\theta_2)p(\text{"text"}|\theta_2)^2 p(\text{"mining"}|\theta_2)^2}$$

$$= \frac{0.5 * 0.5^2 * 0.2^2}{0.5 * 0.5^2 * 0.2^2 + 0.5 * 0.1^2 * 0.1^2} = \frac{100}{101}$$

$$p(Z_d = 2 | d) = ? \quad \frac{p(\theta_2) \cdot p(text|\theta_2)^2 \cdot p(mining|\theta_2)^2}{\text{分母相同}}$$

Normalization to Avoid Underflow

	$p(w\|\theta_1)$	$p(w\|\theta_2)$	$p(w\|\bar{\theta})$
text	0.5	0.1	(0.5+0.1)/2
mining	0.2	0.1	(0.2+0.1)/2
medical	0.2	0.75	(0.2+0.75)/2
health	0.1	0.05	(0.1+0.05)/2

Average of $p(w|\theta_i)$ as a possible normalizer

$$p(Z_d=1|d)=\frac{\dfrac{p(\theta_1)p(\text{"text"}|\theta_1)^2 p(\text{"mining"}|\theta_1)^2}{\boxed{p(\text{"text"}|\bar{\theta})^2 p(\text{"mining"}|\bar{\theta})^2}}}{\dfrac{p(\theta_1)p(\text{"text"}|\theta_1)^2 p(\text{"mining"}|\theta_1)^2}{\boxed{p(\text{"text"}|\bar{\theta})^2 p(\text{"mining"}|\bar{\theta})^2}}+\dfrac{p(\theta_2)p(\text{"text"}|\theta_2)^2 p(\text{"mining"}|\theta_2)^2}{\boxed{p(\text{"text"}|\bar{\theta})^2 p(\text{"mining"}|\bar{\theta})^2}}}$$

An Example of 2 Clusters (cont.)

From E-Step

	$P(Z_d=1\|d)$
d1	0.9
d2	0.1
d3	0.8

	c("text")	c("mining")
d1	2	3
d2	1	2
d3	4	3

M-Step

$p(\theta_1)$=? $p(\theta_2)$= ?

$$p(\theta_1) = \frac{p(Z_{d_1}=1|d_1)+p(Z_{d_2}=1|d_2)+p(Z_{d_3}=1|d_3)}{3}$$

$$= \frac{0.9+0.1+0.8}{3} = 0.6$$

$\therefore p(\theta_2) = 0.4$

	$p(w\|\theta_1)$	$p(w\|\theta_2)$
text	?	?
mining	?	?
medical	?	?
health	?	?

$$p(\text{"text"}|\theta_1) \propto c(\text{"text"},d_1)*p(Z_{d_1}=1|d_1)+\ldots+c(\text{"text"},d_3)*p(Z_{d_3}=1|d_3)$$

$$= 2*0.9+1*0.1+4*0.8$$

$$p(\text{"mining"}|\theta_1) \propto 3*0.9+2*0.1+3*0.8$$

$$p(\text{"text"}|\theta_1)+p(\text{"mining"}|\theta_1)+p(\text{"medical"}|\theta_1)+p(\text{"health"}|\theta_1)=1$$

Summary of Generative Model for Clustering

- A slight variation of topic model can be used for clustering documents
  - Each **cluster** is represented by a **unigram LM** $\mathbf{p(w|\theta_i)}$ ➔ **Term cluster**
  - A document is generated by first choosing a unigram LM and then generating **ALL words** in the document using this **single LM**
  - Estimated model parameters give both a topic characterization of each cluster and a probabilistic assignment of a document into each cluster
  - "Hard" clusters can be obtained by forcing a document into the cluster corresponding to the unigram LM most likely used to generate the document
- EM algorithm can be used to compute the ML estimate
  - Normalization is often needed to avoid underflow