DCMLDA Topic Model

Wenzhe Li

November 14, 2013

1 Introduction

This report illustrates many technical details of the algorithms described in "Accounting for burstiness in Topic Models", by Gabriel and Charles, with the hope that everyone who are interested in can implement their own DCMLDA model, or extend their model using DCMLDA component. Also, as noted in the paper, DCMLDA model can be integrated into any existing topic models.

Briefly speaking, the purpose of DCMLDA model is to capture word burstiness, which means if a term is used once in a document, then it is likely to be used again. In order to capture such phenomena, DCMLDA model use document specific topic-word distributions while document for LDA draws each word from global-wise topic word distribution ϕ.

1.1 Generative Process

The figure below shows the generative process of DCMLDA topic model. As we can see clearly, for each document, we have topic-word distribution. Thus, the difference between LDA and DCMLDA is that DCMLDA pulls from the global-wise to document level.

Algorithm 1 DCMLDA Generative Model

for document d $\in \{1, ..., D\}$ **do**
  draw topic distribution $\theta_d \sim Dir(\alpha)$
  for topic $k \in \{1, ..., K\}$ **do**
    draw topic-word distribution $\phi_{dk} \sim Dir(\beta_k)$
  end for
  for word $n \in \{1, ..., N_d\}$ **do**
    draw topic $z_{dn} \sim \theta_d$
    draw word $w_{dn} \sim \phi_{dz_{dn}}$
  end for
end for

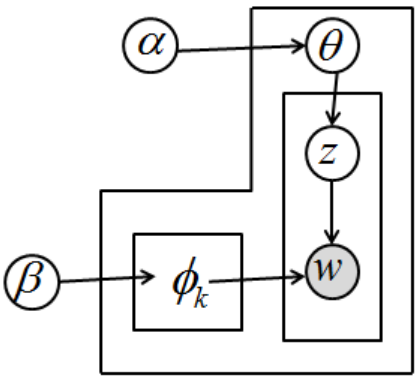

Figure 1: DCMLDA model

2 Derivation of Algorithms

For bayseian models, MCMC sampling is the most common techniques to use for inferring posterior distributions. Here, we use collapsed gibbs sampling to learn the latent variables. As the term "collapsed" means, we integrate over all the parameters except latent variable Z. Next, we show the detailed derivation of collapsed gibbs sampling for DCMLDA model.

2.1 Collapsed Gibbs Sampler

In order to derive the gibbs sampler, we need to calculate $p(z_j|z_{-j}, w)$, and

$$p(z_j|z_{-j}, w) = \frac{p(z, w|\alpha, \beta)}{p(z_{-j}, w|\alpha, \beta)} \tag{1}$$

Here, z_{-j} means excluding the jth variable. We can further factorize the complete likelihood $p(z, w|\alpha, \beta)$ into

$$p(z, w|\alpha, \beta) = p(z|\alpha)p(w|z, \beta) \tag{2}$$

The first probability is the same as in basic LDA model, such that

$$\begin{aligned}
p(z|\alpha) &= \int p(z|\theta)p(\theta|\alpha)d\theta \\
&= \int \prod_{d=1}^{D} \prod_{i:\delta_i=d} \prod_{k=1}^{K} \theta_{dk}^{I_{z_i=k}} \prod_{d=1}^{D} \frac{1}{c(\alpha)} \prod_{k=1}^{K} \theta_{dk}^{\alpha_k-1} d\theta_d \\
&= \prod_{d=1}^{D} \frac{1}{c(\alpha)} \int \prod_{i:\delta_i=d} \prod_{k=1}^{K} \theta_{dk}^{I_{z_i=k}} \prod_{k=1}^{K} \theta_{dk}^{\alpha_k-1} d\theta_d \\
&= \prod_{d=1}^{D} \frac{1}{c(\alpha)} \int \prod_{k=1}^{K} \theta_{dk}^{\sum_{i:\delta_i=d} I_{z_i=k}+\alpha_k-1} d\theta_d \\
&= \prod_{d=1}^{D} \frac{c(\alpha + \sum_{i:\delta_i=d} I_K(z_i))}{c(\alpha)}
\end{aligned}$$

For the second term, we have

$$\begin{aligned}
p(w|z,\beta) &= \int p(w|z,\phi)p(\phi|\beta)d\phi \\
&= \prod_{d=1}^{D} \prod_{k=1}^{K} \prod_{i:z_i=k,\delta_i=d} \prod_{v=1}^{V} \phi_{dkv}^{I_{w_i=v}} \prod_{d=1}^{D} \prod_{k=1}^{K} \frac{1}{c(\beta_k)} \prod_{v=1}^{V} \phi_{dkv}^{\beta_{kv}-1} d\phi_{dk} \\
&= \prod_{d=1}^{D} \prod_{k=1}^{K} \frac{1}{c(\beta_k)} \int \prod_{v=1}^{V} \phi_{dkv}^{\sum_{\substack{i:z_i=k \\ \delta i=d}} I_{w_i=v}+\beta_{kv}-1} d\phi_{dk} \\
&= \prod_{d=1}^{D} \prod_{k=1}^{K} \frac{c(\beta_k + \sum_{\substack{i:z_i=k \\ \delta_i=d}} I_V(w_i))}{c(\beta_k)}
\end{aligned}$$

By combining two terms, we have complete likelihood

$$p(z,w|\alpha,\beta) = \prod_{d=1}^{D} \frac{c(\alpha + \sum_{i:\delta_i=d} I_K(z_i))}{c(\alpha)} \prod_{d=1}^{D} \prod_{k=1}^{K} \frac{c(\beta_k + \sum_{\substack{i:z_i=k \\ \delta_i=d}} I_V(w_i))}{c(\beta_k)} \quad (3)$$

Similarly, we have

$$p(z_{-j},w_{-j}|\alpha,\beta) = \prod_{d=1}^{D} \frac{c(\alpha + \sum_{\substack{i:\delta_i=d \\ i\neq j}} I_K(z_i))}{c(\alpha)} \prod_{d=1}^{D} \prod_{k=1}^{K} \frac{c(\beta_k + \sum_{\substack{i:z_i=k \\ \delta_i=d \\ i\neq j}} I_V(w_i))}{c(\beta_k)} \quad (4)$$

By combining equation (3) and (4),

$$p(z_j|z_{-j},w) = \frac{\prod_{d=1}^{D} \frac{c(\alpha+\sum_{i:\delta_i=d} I_K(z_i))}{c(\alpha)} \prod_{d=1}^{D} \prod_{k=1}^{K} \frac{c(\beta_k+\sum_{\substack{i:z_i=k \\ \delta_i=d}} I_V(w_i))}{c(\beta_k)}}{\prod_{d=1}^{D} \frac{c(\alpha+\sum_{\substack{i:\delta_i=d \\ i\neq j}} I_K(z_i))}{c(\alpha)} \prod_{d=1}^{D} \prod_{k=1}^{K} \frac{c(\beta_k+\sum_{\substack{i:z_i=k \\ \delta_i=d \\ i\neq j}} I_V(w_i))}{c(\beta_k)}}$$

Finally, we have collapsed gibbs sampler as

$$p(z_j = k|z_{-j}, w) = \frac{c(\alpha + \sum_{i:\delta_i=\delta_j, i\neq j} I_K(z_i))}{c(\alpha + \sum_{i:\delta_i=\delta_j, i\neq j} I_K(z_i))} \times \frac{c(\beta_k + \sum_{i:\delta_i=\delta_j, z_i=k, i\neq j} I_V(w_i))}{c(\beta_k + \sum_{i:\delta_i=\delta_j, z_i=k, i\neq j} I_V(w_i))}$$

After replacing C(*) with expanded form (gamma function), we have

$$p(z_j = k|z_{-j}, w) = \frac{\alpha_k + n_{d,k}}{\sum_i \alpha_i + n_d} \times \frac{\beta_{k,w_i} + n_{d,k,w_i}}{\sum_v \beta_{k,v} + n_{d,k}} \tag{5}$$

The meanings of variables are as follows (all are excluding the current word):

- $n_{d,k}$: # of words in document d, that are assigned to topic k.
- n_d: # of words in document d
- n_{d,k,w_i}: # of word w_i, in document d, and assigned to topic k.

2.2 Hyperparameter Estimation

For DCMLDA model, it's very important to estimate the hyperparameters, α and β. Since $p(w|\alpha, \beta)$ is intractable, let's compute $p(w, z|\alpha, \beta)$, and maximize it given z

Re-write equation (3), and replace the with gamma functions,

$$\begin{aligned}
p(z, w|\alpha, \beta) &= \prod_{d=1}^{D} \frac{c(\alpha + \sum_{i:\delta_i=d} I_K(z_i))}{c(\alpha)} \prod_{d=1}^{D}\prod_{k=1}^{K} \frac{c(\beta_k + \sum_{i:z_i=k, \delta_i=d} I_V(w_i))}{c(\beta_k)} \\
&= \prod_{d=1}^{D} \frac{\prod_{k=1}^{K} \Gamma(\alpha_k + n_{d,k})/\Gamma(\sum_{k=1}^{K} n_{d,k} + \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)/\Gamma(\sum_k n_{d,k} + \alpha_k)} \times \\
&\quad \prod_{d=1}^{D}\prod_{k=1}^{K} \frac{\prod_{v=1}^{V} \Gamma(\beta_{k,v} + n_{d,k,v})/\Gamma(\sum_v \beta_{k,v})}{\prod_{v=1}^{V} \Gamma(\beta_{k,v})/\Gamma(\sum_v n_{d,k,v} + \beta_{k,v})}
\end{aligned}$$

By taking log, for the first term we have

$$\begin{aligned}
&\log(\prod_{d=1}^{D} \frac{\prod_{k=1}^{K} \Gamma(\alpha_k + n_{d,k})/\Gamma(\sum_{k=1}^{K} n_{d,k} + \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)/\Gamma(\sum_k n_{d,k} + \alpha_k)}) \\
= &\sum_{d=1}^{D} (\log \prod_{k=1}^{K} \Gamma(\alpha_k + n_{d,k}) \Gamma(\sum_{k=1}^{K} \alpha_k)) - \log(\sum_{k=1}^{K} \Gamma(\alpha_k) \Gamma(\sum_{k=1}^{K} n_{d,k} + \alpha_k)) \\
= &\sum_d \sum_k (\log \Gamma(\alpha_k + n_{d,k}) - \log \Gamma(\alpha_k)) + \sum_d (\log \Gamma(\sum_k \alpha_k) - \log \Gamma(\sum_k n_{d,k} + \alpha_k))
\end{aligned}$$

Similarity, for the second term, we have

$$\begin{aligned} & \log(\prod_{d=1}^{D}\prod_{k=1}^{K}\frac{\prod_{v=1}^{V}\Gamma(\beta_{k,v}+n_{d,k,v})/\Gamma(\sum_v \beta_{k,v})}{\prod_{v=1}^{V}\Gamma(\beta_{k,v})/\Gamma(\sum_v n_{d,k,v}+\beta_{k,v})}) \\ = & \sum_d\sum_k\sum_v(\log\Gamma(\beta_{k,v}+n_{d,k,v})-\log\Gamma(\beta_{k,v}))+ \\ & \sum_d\sum_k(\log\Gamma(\sum_v\beta_{k,v})-\log\Gamma(\sum_v n_{d,k,v}+\beta_{k,v})) \end{aligned}$$

In order to maximize $p(w,z|\alpha,\beta)$, we can maximize following K+1 equations independently.

$$\alpha' = \arg\max_\alpha \sum_d\sum_k(\log\Gamma(\alpha_k+n_{d,k})-\log\Gamma(\alpha_k))+\sum_d(\log\Gamma(\sum_k\alpha_k)-\log\Gamma(\sum_k n_{d,k}+\alpha_k))$$

$$\beta'_k = \arg\max_{\beta_k} \sum_d\sum_k\sum_v(\log\Gamma(\beta_{kv}+n_{d,k,v})-\log\Gamma(\beta_{k,v}))+\sum_d\sum_k(\log\Gamma(\sum_v\beta_{kv})-\log\Gamma(\sum_v n_{d,k,v}+\beta_{kv}))$$

For $k \in [1,...K]$

2.2.1 Solve non-linear optimization directly

L-BFGS is the most common algorithm for solving this problem. There are cool implementation by Dan Klein(java), and Mark Schmidt(matlab).

2.2.2 Solve non-linear optimization using fixed point iteration

In this section, we will solve the problem using fixed point iteration method. If you never heard about this method, then just take 5 minutes to look into the wiki[3], it's very simple. Let's rewrite the optimization for alpha as:

$$f(\alpha) = \sum_{d,k}(\log\Gamma(\alpha_k+n_{d,k})-\log\Gamma(\alpha_k))+\sum_d(\log\Gamma(\sum_k\alpha_k)-\log\Gamma(\sum_k n_{d,k}+\alpha_k)) \tag{6}$$

Then we take partial derivative for α_k, for $k \in [1,...K]$

$$\frac{\partial}{\partial\alpha_k}f(\alpha) = \sum_d(\psi(\alpha_k+n_{d,k})-\psi(\alpha_k))+\sum_d(\psi(\sum\alpha_k)-\psi(n_d+\sum\alpha_k)) \tag{7}$$

Based on fixed point iteration method, we can easily get

$$\alpha'_k = \alpha_k * \frac{\sum_d(\psi(\alpha_k+n_{d,k})-\psi(\alpha_k))}{\sum_d(\psi(n_d+\sum\alpha_k)-\psi(\sum\alpha_k))} \tag{8}$$

One important property that digamma function $\psi(x)$ has is:

$$\psi(x+1)-\psi(x) = \frac{1}{x}$$

Using this property, we can rewrite the function (8) as:

$$\alpha'_k = \alpha_k * \frac{\sum_d \sum_{j=1}^{n_{d,k}-1} \frac{1}{j+\alpha_k}}{\sum_d \sum_{j=0}^{n_d-1} \frac{1}{j+\sum_k \alpha_k}} \tag{9}$$

Similarly, for parameter β, we have:

$$\beta'_{kv} = \frac{\sum_d \sum_{j=0}^{n_{dkv}-1} \frac{1}{j+\beta_{kv}}}{\sum_d \sum_{j=0}^{n_{dk}-1} \frac{1}{j+\beta_k}} \tag{10}$$