

A Tutorial on Probabilistic Latent Semantic Analysis

Liangjie Hong

Department of Computer Science and Engineering

Lehigh University

<http://www.lehigh.edu/~lih307>

hongliangjie@lehigh.edu

November 1, 2010

Abstract

In this tutorial, we would introduce the basic Probabilistic Latent Semantic Analysis model (PLSA) and its variants. This tutorial focuses on detailed derivations of models and their inference algorithms, providing a self-learning material for researchers in information retrieval and social networks. In addition to the basic model, some extensions and variants are also included in the tutorial.

1 Some History

Historically, many believe that these three papers [7, 8, 9] established the techniques of Probabilistic Latent Semantic Analysis or PLSA for short. However, there also exists one variant of the model in [11] and indeed all these models were originally discussed in an earlier technical report [10]. In [2], the authors extended MLE-style estimation of PLSA to MAP-style estimations. A hierarchical extension was proposed in [6]. In [4], the authors showed the equivalent between PLSA and another popular method, non-negative matrix factorization. A high order of proof was shown in [12]. The equivalent between PLSA and LDA was shown in [5].

2 A Modern View of PLSA

In order to better understand the intuition behind the model, we need to make some assumptions. First, we assume a topic ϕ_k is a distribution over a fixed size of vocabulary V . In the original PLSA model, this distribution is not explicitly specified but the form is in Multinomial distribution. Thus, ϕ_k is essentially a vector that each element $\phi_{(k,w)}$ represents the probability that term w is chosen by topic k , namely:

$$p(w|k) = \phi_{(k,w)} \quad (1)$$

and note $\sum_w \phi_{(k,w)} = 1$. Secondly, we also assume that a document consists of multiple topics. Therefore, there is a distribution θ_d over a fixed number of topics T for each document d . Similarly, original PLSA model does not have the explicit specification of this distribution but it is indeed a Multinomial distribution where each element $\theta_{(d,k)}$ in the vector θ_d represents the probability that topic k appears in document d , namely:

$$p(k|d) = \theta_{(d,k)} \quad (2)$$

and also $\sum_k \theta_{(d,k)} = 1$. This is the prerequisite of the model.

PLSA can be considered as a generative model, although it is not strictly the case [1]. Before we start, there is one subtle issue needs to be pointed out. That is the difference between a term w in the

vocabulary V and a token position d_i in a document d . Terms in the vocabulary are distinct, meaning that all the terms differ from each other. Token positions are the places where terms are realized. Therefore, a term could appear multiple times in a same document d in different token positions.

Imagine someone wants to write a document, he needs to decide which term to choose for each token position in a document d . For i -th position, he first decides which topic he wants to write, according to the distribution θ_d . In this step, he essentially flips a T -side dice since θ_d is a Multinomial distribution. Once the outcome of decision is made, suppose it is topic k , he then chooses a term, according to the distribution ϕ_k . Similarly, a V -side dice is flipped. This two step generation process is repeated for all token positions and for all documents in the dataset.

The generation process can be summarized as follows:

- For each document d
 - For each token position i
 - Choose a topic $z \sim \text{Multinomial}(\theta_d)$
 - Choose a term $w \sim \text{Multinomial}(\phi_z)$

and we can write the probability a term w appearing at token position i in document d as follows:

$$p(d_i = w | \Phi, \theta_d) = \sum_{z=k}^T \phi_{(z,w)} \theta_{(d,z)} \quad (3)$$

and the joint likelihood of the whole dataset \mathcal{W} is:

$$\begin{aligned} p(\mathcal{W} | \Phi, \Theta) &= \prod_d^D \prod_i^{N_d} \sum_{z=k}^T \phi_{(z,w)} \theta_{(d,z)} \\ &= \prod_d^D \prod_w^V \left(\sum_{z=k}^T \phi_{(z,w)} \theta_{(d,z)} \right)^{n(d,w)} \end{aligned} \quad (4)$$

where $n(d, w)$ is the number of times term w appearing in document d .

In the formalism above, the likelihood depends on parameters Φ and Θ , which needs to be estimated from data. Here, we wish to obtain the parameters that can maximize the above likelihood. Therefore, we have:

$$\arg \max_{\Phi, \Theta} \left[\log p(\mathcal{W} | \Phi, \Theta) + \sum_d^D \lambda_d \left(1 - \sum_z^T \theta_{(d,z)} \right) + \sum_z^T \sigma_k \left(1 - \sum_w^V \phi_{(z,w)} \right) \right] \quad (5)$$

where the second and the third part of the equation is Lagrange Multipliers to guarantee Multinomial parameters in range $[0, 1]$.

It is difficult to directly optimize the above equation due to the log sign is out of a summation. EM (Expectation Maximization) [3] algorithm is employed here to estimate these parameters. The key assumption to apply EM algorithm is that we know for each token position which topic is chosen from. In other words, for each token position, we know z value. Note, we just *pretend* we know these values. We denote $R_{w_{di}}$ to represent which z is chosen for token position di in document d . Thus, $R_{w_{di}}$ is a T dimensional vector where $\sum_k R_{(w_{di},k)} = 1$. This also indicates that each $R_{w_{di}}$ is in fact a valid distribution and \mathbf{R} is a matrix where each row entry is a $R_{w_{di}}$. We plug all these hidden variables into the likelihood function:

$$\mathcal{L} = \log p(\mathcal{W} | \mathbf{R}, \Phi, \Theta) = \sum_d^D \sum_{di}^{N_d} \sum_z^T R_{(w_{di},z)} \left(\log \phi_{(z,w_{di})} + \log \theta_{(d,z)} \right) \quad (6)$$

and our new objective function is as follows:

$$\arg \max_{\Phi, \Theta} \Lambda = \left[\log p(\mathcal{W} | \mathbf{R}, \Phi, \Theta) + \sum_d^D \lambda_d \left(1 - \sum_z^T \theta_{(d,z)} \right) + \sum_z^T \sigma_z \left(1 - \sum_w^V \phi_{(z,w)} \right) \right] \quad (7)$$

For a standard E-step in EM algorithm, we compute the posterior distribution of hidden variables, given the data and the current values of parameters:

$$\begin{aligned} \langle R_{(w_{di},k)} \rangle &= p(R_{(w_{di},k)} = 1 | \mathcal{W}, \Theta, \Phi) \\ &= \frac{p(\mathcal{W}, R_{(w_{di},k)} = 1 | \Theta, \Phi)}{\sum_k^T p(\mathcal{W}, R_{(w_{di},k)} = 1 | \Theta, \Phi)} \\ &= \frac{p(w_{di}, R_{(w_{di},k)} = 1 | \theta_d, \Phi)}{\sum_k^T p(w_{di}, R_{(w_{di},k)} = 1 | \theta_d, \Phi)} \\ &= \frac{p(w_{di} | \phi_{(k,w_{di})}) p(k | \theta_d)}{\sum_k^T p(w_{di} | \phi_{(k,w_{di})}) p(k | \theta_d)} \\ &= \frac{\phi_{(k,w_{di})} \theta_{(d,k)}}{\sum_k^T \phi_{(k,w_{di})} \theta_{(d,k)}} \end{aligned} \quad (8)$$

In M-step, we obtain the new optimal values for parameters given the current settings of hidden variables. For θ_d , we have:

$$\begin{aligned} \frac{\partial \Lambda}{\partial \theta_{(d,z)}} &= \sum_{di}^{N_d} \frac{\langle R_{(w_{di},z)} \rangle}{\theta_{(d,z)}} - \lambda_d = 0 \\ \frac{\partial \Lambda}{\partial \lambda_d} &= 1 - \sum_z^T \theta_{(d,z)} = 0 \end{aligned} \quad (9)$$

Solving the above two equations, we obtain:

$$\theta_{(d,z)} = \frac{\sum_{di} \langle R_{(w_{di},z)} \rangle}{N_d} \quad (10)$$

Similarly, for ϕ_z , we have:

$$\begin{aligned} \frac{\partial \Lambda}{\partial \phi_{(z,w)}} &= \sum_d^D \sum_{di}^{N_d} \frac{\langle R_{(w_{di},z)} \rangle \mathbb{I}(w_{di} = w)}{\phi_{(z,w)}} - \sigma_z = 0 \\ \frac{\partial \Lambda}{\partial \sigma_z} &= 1 - \sum_w^V \phi_{(z,w)} = 0 \end{aligned} \quad (11)$$

Solving the above two equations, we obtain:

$$\phi_{(z,w)} = \frac{\sum_d^D \sum_{di}^{N_d} \langle R_{(w_{di},z)} \rangle \mathbb{I}(w_{di} = w)}{\sum_{w'}^V \sum_d^D \sum_{di}^{N_d} \langle R_{(w_{di},z)} \rangle \mathbb{I}(w_{di} = w')} \quad (12)$$

Note, we can simplify the notation of EM step. Notice that for all token positions of a same term w in a same document d , E-step is essentially same and therefore simplified E-step is:

$$\langle R_{(w,k)}^{(d)} \rangle = \frac{\phi_{(k,w)} \theta_{(d,k)}}{\sum_k^T \phi_{(k,w)} \theta_{(d,k)}} \quad (13)$$

and simplified M-step is:

$$\begin{aligned}\theta_{(d,k)} &= \frac{\sum_w^V n(d,w) \langle R_{(w,k)}^{(d)} \rangle}{N_d} \\ \phi_{(k,w)} &= \frac{\sum_d^D n(d,w) \langle R_{(w,k)}^{(d)} \rangle}{\sum_{w'}^V \sum_d^D n(d,w') \langle R_{(w',k)}^{(d)} \rangle}\end{aligned}\quad (14)$$

3 Further Discussion on EM Algorithm

In the above discussion, there is one subtle detail that needs more space to be clarified. We introduced $R_{(w_{di},k)}$ as indicator variables to indicate which topic is chosen for token position di . Although it satisfies $\sum_k R_{(w_{di},k)} = 1$, this vector essentially only has one element equal to 1. However, when we calculate E-step of the inference algorithm, we calculate $\langle R_{(w_{di},k)} \rangle$, the posterior distribution of hidden variables, given the data and current settings of parameters. Here, $\langle R_{(w_{di},k)} \rangle$ is a distribution and it has probabilities in each element of the vector but still satisfies $\sum_k \langle R_{(w_{di},k)} \rangle = 1$. What really leads to this difference?

We re-write the log likelihood of one token position after we introduce the indicator variables as follows:

$$\log \sum_k^T R_{(w_{di},k)} \left(\phi_{(k,w_{di})} \theta_{(d,k)} \right) \quad (15)$$

We introduce an auxiliary distribution $q(R_{(w_{di},k)}) = q(R_{(w_{di},k)} = 1)$ and therefore $\sum_k q(R_{(w_{di},k)}) = 1$. Plug this auxiliary distribution into the above log likelihood, we obtain:

$$\log \sum_k^T \frac{R_{(w_{di},k)} \left(\phi_{(k,w_{di})} \theta_{(d,k)} \right)}{q(R_{(w_{di},k)})} q(R_{(w_{di},k)}) = \log \mathbb{E}_q \left[\frac{R_{(w_{di},k)} \left(\phi_{(k,w_{di})} \theta_{(d,k)} \right)}{q(R_{(w_{di},k)})} \right] \quad (16)$$

By using Jensen's Inequality, we can move the log sign into the expectation and make a lower bound of our original log likelihood:

$$\begin{aligned}\log \mathbb{E}_q \left[\frac{R_{(w_{di},k)} \left(\phi_{(k,w_{di})} \theta_{(d,k)} \right)}{q(R_{(w_{di},k)})} \right] &\geq \mathbb{E}_q \left[\log \frac{R_{(w_{di},k)} \left(\phi_{(k,w_{di})} \theta_{(d,k)} \right)}{q(R_{(w_{di},k)})} \right] \\ &\geq \mathbb{E}_q \left[\log \left(R_{(w_{di},k)} \left(\phi_{(k,w_{di})} \theta_{(d,k)} \right) \right) - \log q(R_{(w_{di},k)}) \right]\end{aligned}\quad (17)$$

Now, our goal is clear. Since it is hard to directly optimize the left hand side, we need to maximize the lower bound, right hand side, as much as possible:

$$\sum_k q(R_{(w_{di},k)}) \log \left(R_{(w_{di},k)} \left(\phi_{(k,w_{di})} \theta_{(d,k)} \right) \right) - \sum_k q(R_{(w_{di},k)}) \log q(R_{(w_{di},k)}) + \lambda \left(1 - \sum_k q(R_{(w_{di},k)}) \right) \quad (18)$$

Taking the derivatives respect to $q(R_{(w_{di},k)})$ and setting to 0, we have:

$$\log \left(R_{(w_{di},k)} \left(\phi_{(k,w_{di})} \theta_{(d,k)} \right) \right) - \log q(R_{(w_{di},k)}) - 1 - \lambda = 0 \quad (19)$$

Solving this, we obtain:

$$q(R_{(w_{di},k)}) = \frac{\phi_{(k,w_{di})} \theta_{(d,k)}}{\sum_k^T \phi_{(k,w_{di})} \theta_{(d,k)}} \quad (20)$$

It is exactly E-step we obtained in the previous section. Note, $q(R_{(w_{di},k)})$ is indeed $\langle R_{(w_{di},k)} \rangle$ and we understand that EM algorithm here in a lower bound maximization process.

4 Original Formalism of PLSA

There are two ways to formulate PLSA. They are equivalent but may lead to different inference process.

$$P(d, w) = P(d) \sum_z P(w|z)P(z|d) \quad (21)$$

$$P(d, w) = \sum_z P(w|z)P(d|z)P(z) \quad (22)$$

Let's see why these two equations are equivalent by using Bayes rule.

$$\begin{aligned} P(z|d) &= \frac{P(d|z)P(z)}{P(d)} \\ P(z|d)P(d) &= P(d|z)P(z) \\ P(w|z)P(z|d)P(d) &= P(w|z)P(d|z)P(z) \\ P(d) \sum_z P(w|z)P(z|d) &= \sum_z P(w|z)P(d|z)P(z) \end{aligned}$$

The whole data set is generated as (we assume that all words are generated independently):

$$D = \prod_d \prod_w P(d, w)^{n(d, w)} \quad (23)$$

The Log-likelihood of the whole data set for (1) and (2) are:

$$L_1 = \sum_d \sum_w n(d, w) \log[P(d) \sum_z P(w|z)P(z|d)] \quad (24)$$

$$L_2 = \sum_d \sum_w n(d, w) \log[\sum_z P(w|z)P(d|z)P(z)] \quad (25)$$

5 EM

For L_1 or L_2 , the optimization is hard due to the log of sum. Therefore, an algorithm called Expectation-Maximization is usually employed. Before we introduce anything about EM, please note that EM is only guarantee to find a local optimum (although it may be a global one).

First, we see how EM works in general. As we shown for PLSA, we usually want to estimate the likelihood of data, namely $P(X|\theta)$, given the paramter θ . The easiest way is to obtain a maximum likelihood estimator by maximizing $P(X|\theta)$. However, sometimes, we also want to include some hidden variables which are usually useful for our task. Therefore, what we really want to maximize is $P(X|\theta) = \sum_z P(X|z, \theta)P(z|\theta)$, the complete likelihood. Now, our attention becomes to this complete likelihood. Again, directly maximizing this likelihood is usually difficult. What we would like to show here is to obtain a lower bound of the likelihood and maximize this lower bound.

We need Jensen's Inequality to help us obtain this lower bound. For any convex function $f(x)$, Jensen's Inequality states that :

$$\lambda f(x) + (1 - \lambda)f(y) \geq f(\lambda x + (1 - \lambda)y)$$

Thus, it is not difficult to show that :

$$E[f(x)] = \sum_x P(x)f(x) \geq f(\sum_x P(x)x) = f(E[x])$$

For concave functions (like logarithm), it is :

$$E[f(x)] \leq f(E[x])$$

Back to our complete likelihood, we can obtain the following conclusion by using concave version of Jensen's Inequality :

$$\begin{aligned} \log \sum_z P(X|z, \theta)P(z|\theta) &= \log \sum_z P(X|z, \theta)P(z|\theta) \frac{q(z)}{q(z)} \\ &= \log E\left[\frac{P(X|z, \theta)P(z|\theta)}{q(z)}\right] \\ &\geq E\left[\log \frac{P(X|z, \theta)P(z|\theta)}{q(z)}\right] \end{aligned}$$

Therefore, we obtained a lower bound of complete likelihood and we want to maximize it as tight as possible. EM is an algorithm that maximize this lower bound through a iterative fashion. Usually, EM first would fix current θ value and maximize $q(z)$ and then use the new $q(z)$ value to obtain a new guess on θ , which is essentially a two stage maximization process. The first step can be shown as follows:

$$\begin{aligned} E\left[\log \frac{P(X|z, \theta)P(z|\theta)}{q(z)}\right] &= \sum_z q(z) \log \frac{P(X|z, \theta)P(z|\theta)}{q(z)} \\ &= \sum_z q(z) \log \frac{P(z|X, \theta)P(X, \theta)}{q(z)} \\ &= \sum_z q(z) \log P(x, \theta) + \sum_z q(z) \log \frac{P(z|X, \theta)}{q(z)} \\ &= \log P(x, \theta) - \sum_z q(z) \log \frac{q(z)}{P(z|X, \theta)} \\ &= \log P(x, \theta) - E\left[\log \frac{q(z)}{P(z|X, \theta)}\right] \\ &= \log P(x, \theta) - KL(q(z)||P(z|X, \theta)) \end{aligned}$$

The first term is the same for all z . Therefore, in order to maximize the whole equation, we need to minimize KL divergence between $q(z)$ and $P(z|X, \theta)$, which eventually leads to the optimum solution of $q(z) = P(z|X, \theta)$. So, usually for E-step, we use current guess of θ to calculate the posterior distribution of hidden variable as the new update score. For M-step, it is problem-dependent. We will see how to do that in later discussions.

We also show another explanation of EM in terms of optimizing a so-called Q function. We devise the data generation process as $P(X|\theta) = P(X, H|\theta) = P(H|X, \theta)P(X|\theta)$. Therefore, the complete likelihood is modified as:

$$L_c(\theta) = \log P(X, H|\theta) = \log P(X|\theta) + \log P(H|X, \theta) = L(\theta) + \log P(H|X, \theta)$$

Think about how to maximize $L_c(\theta)$. Instead of directly maximizing it, we can iteratively maximize $L_c(\theta^{(n+1)}) - L_c(\theta^{(n)})$ as :

$$\begin{aligned} L(\theta) - L(\theta^{(n)}) &= L_c(\theta) - \log P(H|X, \theta) - L_c(\theta^{(n)}) + \log P(H|X, \theta^{(n)}) \\ &= L_c(\theta) - L_c(\theta^{(n)}) + \log \frac{P(H|X, \theta^{(n)})}{P(H|X, \theta)} \end{aligned}$$

Now take the expectation of this equation, we have:

$$L(\theta) - L(\theta^{(n)}) = \sum_H L_c(\theta)P(H|X, \theta^{(n)}) - \sum_H L_c(\theta^{(n)})P(H|X, \theta^{(n)}) + \sum_H P(H|X, \theta^{(n)}) \log \frac{P(H|X, \theta^{(n)})}{P(H|X, \theta)}$$

The last term is always non-negative since it can be recognized as the KL-divergence of $P(H|X, \theta^{(n)})$ and $P(H|X, \theta)$. Therefore, we obtain a lower bound of Likelihood :

$$L(\theta) \geq \sum_H L_c(\theta)P(H|X, \theta^{(n)}) + L(\theta^{(n)}) - \sum_H L_c(\theta^{(n)})P(H|X, \theta^{(n)})$$

The last two terms can be treated as constants as they do not contain the variable θ , so the lower bound is essentially the first term, which is also sometimes called as ‘‘Q-function’’.

$$Q(\theta; \theta^{(n)}) = E(L_c(\theta)) = \sum_H L_c(\theta)P(H|X, \theta^{(n)}) \quad (26)$$

5.1 EM of Formulation 1

In case of Formulation 1, let us introduce hidden variables $R(z, w, d)$ to indicate which hidden topic z is selected to generated w in d ($\sum_z R(z, w, d) = 1$). Therefore, the complete likelihood can be formulated as :

$$\begin{aligned} L_{c1} &= \sum_d \sum_w n(d, w) \sum_z R(z, w, d) \log[P(d)P(w|z)P(z|d)] \\ &= \sum_d \sum_w n(d, w) \sum_z R(z, w, d) [\log P(d) + \log P(w|z) + \log P(z|d)] \end{aligned}$$

From the equation above, we can write our Q-function for the complete likelihood $E[L_{c1}]$:

$$E[L_{c1}] = \sum_d \sum_w n(d, w) \sum_z P(z|w, d) [\log P(d) + \log P(w|z) + \log P(z|d)]$$

For E-step, simply using Bayes Rule, we can obtain:

$$\begin{aligned} P(z|w, d) &= \frac{P(w|z, d)}{P(w, d)} \\ &= \frac{P(w|z)P(z|d)P(d)}{\sum_z P(w|z)P(z|d)P(d)} \\ &= \frac{P(w|z)P(z|d)}{\sum_z P(w|z)P(z|d)} \end{aligned}$$

For M-step, we need to maximize Q-function, which needs to be incorporated with other constraints:

$$H = E[L_{c1}] + \alpha[1 - \sum_d P(d)] + \beta \sum_z [1 - \sum_w P(w|z)] + \gamma \sum_d [1 - \sum_z P(z|d)]$$

and take all derivatives:

$$\begin{aligned} \frac{\partial H}{\partial P(d)} &= \sum_w \sum_z n(d, w) \frac{P(z|w, d)}{P(d)} - \alpha = 0 \\ &\rightarrow \sum_w \sum_z n(d, w) P(z|w, d) - \alpha P(d) = 0 \\ \frac{\partial H}{\partial P(w|z)} &= \sum_d n(d, w) \frac{P(z|w, d)}{P(w|z)} - \beta = 0 \\ &\rightarrow \sum_d n(d, w) P(z|w, d) - \beta P(w|z) = 0 \\ \frac{\partial H}{\partial P(z|d)} &= \sum_w n(d, w) \frac{P(z|w, d)}{P(z|d)} - \gamma = 0 \\ &\rightarrow \sum_w n(d, w) P(z|w, d) - \gamma P(z|d) = 0 \end{aligned}$$

Therefore, we can easily obtain:

$$\begin{aligned}
P(d) &= \frac{\sum_w \sum_z n(d, w) P(z|w, d)}{\sum_d \sum_w \sum_z n(d, w) P(z|w, d)} \\
&= \frac{n(d)}{\sum_d n(d)} \\
P(w|z) &= \frac{\sum_d n(d, w) P(z|w, d)}{\sum_w \sum_d n(d, w) P(z|w, d)} \\
P(z|d) &= \frac{\sum_w n(d, w) P(z|w, d)}{\sum_z \sum_w n(d, w) P(z|w, d)} \\
&= \frac{\sum_w n(d, w) P(z|w, d)}{n(d)}
\end{aligned}$$

5.2 EM of Formulation 2

Use similar method to introduce hidden variables to indicate which z is selected to generated w and d and we can have the following complete likelihood :

$$\begin{aligned}
L_{c2} &= \sum_d \sum_w n(d, w) \sum_z R(z, w, d) \log[P(z)P(w|z)P(d|z)] \\
&= \sum_d \sum_w n(d, w) \sum_z R(z, w, d) [\log P(z) + \log P(w|z) + \log P(d|z)]
\end{aligned}$$

Therefore, the Q-function $E[L_{c2}]$ would be :

$$E[L_{c2}] = \sum_d \sum_w n(d, w) \sum_z P(z|w, d) [\log P(z) + \log P(w|z) + \log P(d|z)]$$

For E-step, again, simply using Bayes Rule, we can obtain:

$$\begin{aligned}
P(z|w, d) &= \frac{P(w|z, d)}{P(w, d)} \\
&= \frac{P(w|z)P(d|z)P(z)}{\sum_z P(w|z)P(d|z)P(z)}
\end{aligned}$$

For M-step, we maximize the constraint version of Q-function:

$$H = E[L_{c2}] + \alpha [1 - \sum_z P(z)] + \beta \sum_z [1 - \sum_w P(w|z)] + \gamma \sum_z [1 - \sum_d P(d|z)]$$

and take all derivatives:

$$\begin{aligned}
\frac{\partial H}{\partial P(z)} &= \sum_d \sum_w n(d, w) \frac{P(z|w, d)}{P(z)} - \alpha = 0 \\
&\rightarrow \sum_d \sum_w n(d, w) P(z|w, d) - \alpha P(z) = 0 \\
\frac{\partial H}{\partial P(w|z)} &= \sum_d n(d, w) \frac{P(z|w, d)}{P(w|z)} - \beta = 0 \\
&\rightarrow \sum_d n(d, w) P(z|w, d) - \beta P(w|z) = 0 \\
\frac{\partial H}{\partial P(d|z)} &= \sum_w n(d, w) \frac{P(z|w, d)}{P(d|z)} - \gamma = 0 \\
&\rightarrow \sum_w n(d, w) P(z|w, d) - \gamma P(d|z) = 0
\end{aligned}$$

Therefore, we can easily obtain:

$$\begin{aligned}
P(z) &= \frac{\sum_d \sum_w n(d, w) P(z|w, d)}{\sum_d \sum_w \sum_z n(d, w) P(z|w, d)} \\
&= \frac{\sum_d \sum_w n(d, w) P(z|w, d)}{\sum_d \sum_w n(d, w)} \\
P(w|z) &= \frac{\sum_d n(d, w) P(z|w, d)}{\sum_w \sum_d n(d, w) P(z|w, d)} \\
P(d|z) &= \frac{\sum_w n(d, w) P(z|w, d)}{\sum_d \sum_w n(d, w) P(z|w, d)}
\end{aligned}$$

6 Incorporating Background Language Model

Another PLSA model which incorporates background language model is usually formulated like this :

$$P(d, w) = \lambda_B P(w|\theta_B) + (1 - \lambda_B) \sum_z P(w|z) P(z|d) P(d) \quad (27)$$

The log likelihood of Equation 7 is

$$L = \sum_d \sum_w n(d, w) \log[\lambda_B P(w|\theta_B) + (1 - \lambda_B) \sum_z P(w|z) P(z|d) P(d)]$$

Let's again introduce a hidden variable $P(Z_{d,w})$ to indicate which component that the w and d are generated while $P(Z_{d,w} = \theta_B)$ means that the word is generated by the background model and $P(Z_{d,w} = j)$ meaning the word is generated by the topic z_j . Thus, the complete log likelihood is :

$$L_c = \sum_d \sum_w n(d, w) [P(Z_{d,w} = \theta_B) \log(\lambda_B P(w|\theta_B)) + \sum_z P(Z_{d,w} = z | Z_{d,w} \neq \theta_B) \log((1 - \lambda_B) P(w|z) P(z|d) P(d))]$$

The E-step is straightforward. Using Bayes Rule, we can obtain:

$$\begin{aligned}
P(Z_{d,w} = \theta_B | d, w) &= \frac{P(w|\theta_B, d)}{P(w, d)} \\
&= \frac{\lambda_B P(w|\theta_B)}{\lambda_B P(w|\theta_B) + (1 - \lambda_B) \sum_z P(w|z) P(z|d) P(d)} \\
P(Z_{d,w} = z | d, w) &= \frac{P(w|z, d)}{P(w, d)} \\
&= \frac{P(w|z) P(z|d) P(d)}{\sum_z P(w|z) P(z|d) P(d)} \\
&= \frac{P(w|z) P(z|d)}{\sum_z P(w|z) P(z|d)}
\end{aligned}$$

For M-step, we maximize the constraint version of Q-function:

$$H = E[L_c] + \beta [1 - \sum_w P(w|z)] + \gamma [1 - \sum_z P(z|d)]$$

and take all derivatives:

$$\begin{aligned}
\frac{\partial H}{\partial P(w|z)} &= \sum_d n(d, w) \frac{P(Z_{d,w} = z)}{P(w|z)} - \beta = 0 \\
&\rightarrow \sum_d n(d, w) P(Z_{d,w} = z) - \beta P(w|z) = 0 \\
\frac{\partial H}{\partial P(z|d)} &= \sum_w n(d, w) \frac{P(Z_{d,w} = z)}{P(z|d)} - \gamma = 0 \\
&\rightarrow \sum_w n(d, w) P(Z_{d,w} = z) - \gamma P(z|d) = 0
\end{aligned}$$

Therefore, we can easily obtain:

$$\begin{aligned}
P(w|z) &= \frac{\sum_d n(d, w)(1 - P(Z_{d,w} = \theta_B|d, w))P(Z_{d,w} = z)}{\sum_w \sum_d n(d, w)(1 - P(Z_{d,w} = \theta_B|d, w))P(Z_{d,w} = z)} \\
P(z|d) &= \frac{\sum_w n(d, w)(1 - P(Z_{d,w} = \theta_B|d, w))P(Z_{d,w} = z)}{\sum_z \sum_w n(d, w)(1 - P(Z_{d,w} = \theta_B|d, w))P(Z_{d,w} = z)}
\end{aligned}$$

Note, $P(w|\theta_B)$ is only sampled once by using the equation:

$$P(w|\theta_B) = \frac{\sum_d n(d, w)}{\sum_w \sum_d n(d, w)}$$

If we change to the PLSA Formulation 2, we will get the following E steps:

$$\begin{aligned}
P(Z_{d,w} = \theta_B|d, w) &= \frac{P(w|\theta_B, d)}{P(w, d)} \\
&= \frac{\lambda_B P(w|\theta_B)}{\lambda_B P(w|\theta_B) + (1 - \lambda_B) \sum_z P(w|z)P(d|z)P(z)} \\
P(Z_{d,w} = z|d, w) &= \frac{P(w|z, d)}{P(w, d)} \\
&= \frac{P(w|z)P(d|z)P(z)}{\sum_z P(w|z)P(d|z)P(z)}
\end{aligned}$$

and corresponding M steps:

$$\begin{aligned}
P(w|z) &= \frac{\sum_d n(d, w)(1 - P(Z_{d,w} = \theta_B|d, w))P(Z_{d,w} = z)}{\sum_w \sum_d n(d, w)(1 - P(Z_{d,w} = \theta_B|d, w))P(Z_{d,w} = z)} \\
P(d|z) &= \frac{\sum_w n(d, w)(1 - P(Z_{d,w} = \theta_B|d, w))P(Z_{d,w} = z)}{\sum_d \sum_w n(d, w)(1 - P(Z_{d,w} = \theta_B|d, w))P(Z_{d,w} = z)} \\
P(z) &= \frac{\sum_d \sum_w n(d, w)(1 - P(Z_{d,w} = \theta_B|d, w))P(Z_{d,w} = z)}{\sum_d \sum_w \sum_z n(d, w)(1 - P(Z_{d,w} = \theta_B|d, w))P(Z_{d,w} = z)}
\end{aligned}$$

References

- [1] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *The Journal of Machine Learning Research*, 3:993–1022, 2003.
- [2] J. T. Chien, J. T. Chien, M. S. Wu, and M. S. Wu. Adaptive bayesian Latent Semantic Analysis. *Audio, Speech, and Language Processing, IEEE Transactions on [see also Speech and Audio Processing, IEEE Transactions on]*, 16(1):198–207, 2008.

- [3] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.
- [4] C. Ding, T. Li, and W. Peng. On the equivalence between Non-negative Matrix Factorization and Probabilistic Latent Semantic Indexing. *Computational Statistics & Data Analysis*, 52(8):3913–3927, 2008.
- [5] M. Girolami and A. Kabán. On an equivalence between PLSI and LDA. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 433–434, New York, NY, USA, 2003. ACM.
- [6] T. Hofmann. The cluster-abstraction model: unsupervised learning of topic hierarchies from text data. In *IJCAI'99: Proceedings of the 16th international joint conference on Artificial intelligence*, pages 682–687, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc.
- [7] T. Hofmann. Probabilistic Latent Semantic Analysis. In *UAI '99: Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, pages 289–296, 1999.
- [8] T. Hofmann. Probabilistic Latent Semantic Indexing. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57, New York, NY, USA, 1999. ACM.
- [9] T. Hofmann. Unsupervised learning by Probabilistic Latent Semantic Analysis. *Machine Learning*, 2001.
- [10] T. Hofmann and J. Puzicha. Unsupervised learning from dyadic data. Technical report, MIT, 1998.
- [11] T. Hofmann, J. Puzicha, and M. I. Jordan. Learning from dyadic data. In *Advances in Neural Information Processing Systems 11*, 1999.
- [12] W. Peng. Equivalence between nonnegative tensor factorization and tensorial probabilistic latent semantic analysis. In *SIGIR '09: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 668–669, New York, NY, USA, 2009. ACM.