TWLDA (Term Weighting LDA)

TWLDA is a new topic of LDA which assigns low weights to words with low topic discriminating power. For more details, please refer to "Exploring Topic Discriminating Power of Words in Latent Dirichlet Allocation" created by Kai Yang, Yi Cai and Zhenhong Chen.

MCMC sample algorithm

Markov chain Monte Carlo

Metropolis-Hastings algorithm

Metropolis–Hastings algorithm

Unigram Model

$\vec{p} = (p_1, p_2, ..., p_v)$ refers to the probability of each word to be chosed.
So the result is $w \sim Mult(w|\vec{p})$
For a doc created using Unigram Model, the probability that the doc to be created is

$$p(\vec{w}) = p(w_1, w_2, ..., w_n) = p(w_1)p(w_2)...p(w_n)$$

And for the hypothesis that each doc is independent, a corpus with some doc $W = (\vec{w_1}), (\vec{w_2}), ..., (\vec{w_m})$, the probability is

$$p(W) = p(\vec{w_1})p(\vec{w_2})...p(\vec{w_m})$$

Assume there are N words in the corpus, for each word v_i has appeard n_i times, $\vec{n} = (n_1, n_2, ..., n_v)$ will be a Multinomial Distribution

$$p(\vec{n}) = Mult(\vec{n}|\vec{p}, N) = \binom{N}{\vec{n}} \prod_{k=1}^{V} p_k^{nk}$$

Now the propability of corpus is

$$p(W) = p(\vec{w_1}\vec{w_2}...\vec{w_m}) = \prod_{k=1}^{V} p_k^{nk}$$

So

$$\hat{p}_i = \frac{n_i}{N}$$

Bayes Unigram Model

$$\hat{p}_i = \frac{n_i + \alpha_i}{\Sigma_{i=1}^{V}(n_i + \alpha_i)}$$

LDA Model

Two most important formula

$$\vec{\alpha} \rightarrow \vec{\theta}_m \rightarrow z_{m,n}$$

$$\vec{\beta} \rightarrow \vec{\varphi}_k \rightarrow w_{m,n}|k = z_{m,n}$$

Gibbs Sampling

$$p(z_i = k|\vec{z}_{\neg i}, \vec{w}) \propto \frac{n_{m,\neg i}^{(k)} + \alpha_k}{\Sigma_{k=1}^{K}(n_{m,\neg i}^{(t)} + \alpha_k)} \cdot \frac{n_{k,\neg i}^{(t)} + \beta_t}{\Sigma_{t=1}^{V}(n_{k,\neg i}^{(t)} + \beta_t)}$$

TWLDA

Step1: $\vec{\varphi'} \leftarrow TopicModel()$

In Step 1, a topic model is executed. Then a topic-word distribution $\vec{\varphi'}$ is generated by this topic model.

Step2: $\vec{\sigma} \leftarrow Calculate(\vec{\varphi'})$

In Step2, according to the $\vec{\varphi'}$, we apply a supervised term weighting scheme to calculate weights of word $\vec{\sigma}$. Since supervised schemes have the ability to measure the topic discriminating power of words, in principle, all the supervised term weighting schemes can be applied here.

Step3: Discounting the number of words

Step3 is to discount the number of words by their weights. The number of words is diminished proportionally accoding to weights of words. Hence, the total discounted number of words in document m under topic k is calculated as follow:

$$n'^{(k)}_{m} = \sum_{t=1}^{V} \sigma_t n_{mkt}$$

where σ_t denates the weight of word t, which is ranging from 0 to 1. n_{nkt} is the number of word t belonging to topic k in document m. Similarly, the total discounted number of word t under topic k is calculated as follows:

$$n'^{(t)}_{k} = \sum_{m=1}^{M} \sigma_t n_{mkt}$$

Step4: Executing xLDA with the discounted values

Step4 is to excute the standard LDA or its variants, denoted as xLDA, using the discounted calues calculated in Equations 1 and 2.