CSE 6240
HW4

Ke Wang
903058889

1. Gibbs Sampling

$$P^*(z)\,T(z, z') = P^*(z_i, u_i)\,P^*(z'_{\neq i} \mid u_i)$$
$$= P^*(z_i \mid u_i)\,P^*(u_i)\,P^*(z'_{\neq i} \mid u_i)$$
$$= P^*(z_i \mid u_i)\,P^*(z'_{\neq i}, u_i)$$
$$= P^*(z')\,T(z', z)$$

($P^*(z_i \mid u_i)$ is the probability that random variable ~~Z= has a~~ $Z_i$ has value $z_i'$ given that $Z_j = z_j$ for all $j$ not equal to $i$)

2. Let's say that there are two documents given below focusing on pets:

$d_1$: The Welsh corgi is a small type of herding dog that originated in Wales

$d_2$: A husky and corgi make your heart melt with what we like to call the "Horgi"

query term: husky corgi

mixed with $\lambda = 1/2$

$$P(q|d_1) = \frac{1}{2}\left(\frac{0}{14} + \frac{1}{30}\right) \times \frac{1}{2}\left(\frac{1}{14} + \frac{2}{30}\right) = 0.00115$$

$$P(q|d_2) = \frac{1}{2}\left(\frac{1}{16} + \frac{1}{30}\right) \times \frac{1}{2}\left(\frac{1}{16} + \frac{2}{30}\right) = 0.00309$$

From the results, we can get that $d_2$ is better than $d_1$ given the query. But $d_1$ is still better than other documents focused on computer science (say one document $d_3$ focused on Java. $P(q|d_3) = 0$).

If we don't use mixture model, then $P(q|d_1) = 0$ because "husky" doesn't appear in $d_1$, however, $d_1$ is certainly related to the query.

This mixture can smooth probabilities in document language model: to discount non-zero probabilities and to give some probability mass to unseen words.

CSE 6240
HW 4

Ke Wang
903058889

3. Pagerank and HITS

First, I'll discuss the pros and cons about PageRank and HITS and then I'll compare these two algorithms.

PageRank - Advantages

① PageRank is global measure and query independent
② robust (somewhat) against spam because it's hard to add links to a spam page from other important pages

PageRank - Disadvantages

① PageRank favors older pages because a new page will not have many links
② Google PageRank algorithm is patented.

HITS - Advantages

① has the ability to rank pages according to the query topic.
② can find multi-lingual results
③ has no computational burden on small graphs

HITS - Disadvantages

① query dependent, so the result must be computed on the fly.
② the real Web is not hubs and authorities
③ very easy to spam (by out links)

Difference between PageRank and HITS

① PageRank is query independent
HITS is query dependent
② HITS is more prone to spams. one page can have lots of out links to authorities to pretend to be a good hub
③ PageRank operates on the whole Web graph
HITS operates on a small subgraph (the seed) from the Web graph.
④ the result of HITS is less relevant than PageRank
(from "Comparative Anaysis of PageRank and HITS Algorithms")

CSE 6240
HW4

Ke Wang
9030 58889

4. pLSI

a) Given P(d), P(z|d), P(w|z), we can generate a word from a document.
First, a document is given by P(d)
Then, we generate a topic from the document by P(z|d)
At last, we generate a word from the topic by P(w|z)

b) EM algorithm is used in PLSA/PLSI to learn the (odd+me) parameters because explicitly finding the parameters is hard.
Let the likelihood be $l(\theta)$ where $\theta$ is parameter. Maximizing $l(\theta)$ explicitly might be difficult, so the strategy is to repeatedly construct a lower-bound on $l$ (E-step), and then optimizes that lower bound (M-step)
Specifically, in the Expectation step, calculate the expected value of log likelihood function; in the Maximization step, find the parameter that maximizes the quantity.

c) pLSI can provide p(w|z) and p(z|d),
from the training data and EM algorithm, we can finally get the parameters to calculate p(w,d)

For two documents $d_1$ and $d_2$, we can calculate $P(d_1|z)$ and $P(d_2|z)$ (these can be calculated by the results of EM algorithm)
Then we use $\sum_z P(d_1|z) \cdot P(d_2|z)$ to calculate the distance between the two documents.

CSE 6240
HW4

Ke Wang
903058889

5. LDA

a) $w$: words word-level
$z$: topics corpus-level
$\theta$: documents doc-level
$\alpha$: parameters ~~over~~ the document distribution over topics
to assign probabilities to new documents (somewhat doc-level)
$\beta$: parameters over the topic distribution of words
to assign probabilities to new words (somewhat word-level)

b) E-step: for each document, running the following iterative algorithm to find the optimizing values of the variational parameters:

1. initialize $\phi$ and $\gamma$
2. while not converge
3. for $n = 1$ to $N_d$
4. update $\phi$
5. normalize $\phi$
6. update $\gamma$

M-step: maximize the resulting lower bound on the log likelihood with respect to the model parameters $\alpha$ and $\beta$.