**CS510: Advanced Topics in Information Retrival** **Fall 2017**

# Assignment 6

Zhuo Li(zhuol2) *Handed In: Octorber 26, 2017*

1. Mixture Models

   a. We have

$$\begin{aligned} p(w = the) &= p(w = the|H)p(H) + p(w = the|T)p(T) \\ &= 0.3 \times 0.8 + 0.3 \times 0.2 \\ &= 0.3 \end{aligned}$$

   b. Because each word is written independently, so we still have the probability as 0.3.

$$\begin{aligned} p(w = the) &= p(w = the|H)p(H) + p(w = the|T)p(T) \\ &= 0.3 \times 0.8 + 0.3 \times 0.2 \\ &= 0.3 \end{aligned}$$

   c. According to Bayes rule

$$\begin{aligned} p(H|w = data) &= \frac{p(w = data|H)p(H)}{p(w = data)} \\ &= \frac{p(w = data|H)p(H)}{p(w = data|H)p(H) + p(w = data|T)p(T)} \\ &= \frac{0.1 \times 0.8}{0.1 \times 0.8 + 0.1 \times 0.2} \\ &= 0.8 \end{aligned}$$

   d. The probabilities for each of the five words to be written are

$$\begin{aligned} p(w = the) &= p(w = the|H)p(H) + p(w = the|T)p(T) \\ &= 0.3 \\ p(w = computer) &= p(w = computer|H)p(H) + p(w = computer|T)p(T) \\ &= 0.12 \\ p(w = data) &= p(w = data|H)p(H) + p(w = data|T)p(T) \\ &= 0.1 \\ p(w = baseball) &= p(w = baseball|H)p(H) + p(w = baseball|T)p(T) \\ &= 0.18 \\ p(w = game) &= p(w = game|H)p(H) + p(w = game|T)p(T) \\ &= 0.18 \end{aligned}$$

   *data* is of the least probability to be written and each word is written independently. So *data* should occur least frequently in the paper.

e. Let $c(w)$ to be the frequency of a word $w$ that occurs in the paper $D$. So we can have

$$\begin{aligned} p(w = computer|H) &= \frac{c(w = computer)}{|D|} \\ &= 3/10 \\ p(w = game|H) &= \frac{c(w = game)}{|D|} \\ &= 2/10 \end{aligned}$$

2. EA Algorithm

a. The formula can be the following

$$p(w_i|D_2) = \lambda p(w_i|\theta_1) + (1 - \lambda)p(w_i|C)$$

b. The likelihood can be

$$\begin{aligned} p(D_2|\Lambda) &= \prod_{i=1}^{|D_2|} p(x_i|\Lambda) \\ &= \prod_{i=1}^{|D_2|} [\lambda p(x_i|\theta_1) + (1 - \lambda)p(x_i|C)] \\ &= \prod_{i=1}^{k} [\lambda p(w_i|\theta_1) + (1 - \lambda)p(w_i|C)]^{c(w_i, D_2)} \end{aligned}$$

So log-likelihood is

$$LL(\lambda) = \sum_{i=1}^{k} c(w_i, D_2) \log [\lambda p(w_i|\theta_1) + (1 - \lambda)p(w_i|C)]$$

By introducing hidden variables, we assume

$$z_i = \begin{cases} 0, \text{ if } w_i \text{ from } D_1 \\ 1, \text{ if } w_i \text{ from background} \end{cases}$$

Then the complete data log-likelihood is

$$LL(\lambda) = \sum_{i=1}^{k} c(w_i, D_2) \log [(1 - z_i)\lambda p(w_i|\theta_1) + z_i(1 - \lambda)p(w_i|C)]$$

c. There are $k$ binary hidden variables. We have $k$ words in $D_2$, then the $k$ hidden variables are $z_1,\ z_2,\ ...\ z_k$.

d. We have

$$\begin{aligned} Q(\lambda, \lambda^{(n)}) &= E[LL(\lambda)] \\ &= \sum_{i=1}^{k} c(w_i, D_2) \log[p(z_i = 0|D_2, \lambda^{(n)})\lambda^{(n)} p(w_i|\theta_1) \\ &\quad + p(z_i = 1|D_2, \lambda^{(n)})(1 - \lambda^{(n)}) p(w_i|C)] \end{aligned} \tag{1}$$

The E-step

$$p(z_i = 0|D_2, \lambda^{(n)}) = \frac{\lambda^{(n)} p(w|\theta_1)}{\lambda^{(n)} p(w|\theta_1) + (1 - \lambda^{(n)}) p(w|C)}$$

The M-step
We have $Q(\lambda, \lambda^{(n)})$ in (1), so

$$\frac{\partial Q}{\partial \lambda} = \sum_{i=1}^{k} c(w_i, D_2)\big(\frac{p(z_i = 0|D_2, \lambda^{(n)}}{\lambda} - \frac{p(z_i = 1|D_2, \lambda^{(n)}}{1 - \lambda}\big) = 0$$

then

$$\lambda^{(n+1)} = \frac{\sum_{i=1}^{k} p(z_i = 0|D_2, \lambda^{(n)}) c(w_i, D_2)}{k}$$

3. PLSA (Open Research Question)

a. Yes, because we can focus on two topics (i.e. Chicago and Seattle) and let PLSA to generate the distributions of the two topics and coverage of each for the whole corpus.

b. Yes. For each word appears in sentences with Chicago or Seattle, we can estimate maximum likelihood

$$\begin{aligned} p(w_i|Chicago) &= \frac{c(w_i, \text{Chicago sentences})}{\text{total words in Chicago sentences}} \\ p(w_i|Seattle) &= \frac{c(w_i, \text{Seattle sentences})}{\text{total words in Seattle sentences}} \end{aligned} \tag{2}$$

Then we use EM-algorithm with prior for PLSA with prior $p(w_i|Chicago)$ and $p(w_i|Seattle)$ for prior parameters and $\mu$, where

$$\mu = \text{total words in Chicago sentences} + \text{total words in Seattle sentences}$$

c. We can have

i. Calculate prior parameters in (2)
ii. Do EM-algorithm with prior for PLSA with prior parameters calculated in step i until converge
iii. Classify each sentence with the parameters we get from step ii

d. We can manually label some sentences as test data and use them to test accuracy of our classifier.

e. If some cases have word Chicago but they are actually about Seattle or vice versa, the performance of our algorithm may be worse than average. We can assume that all the manually labeled sentences (i.e. with word Chicago or Seattle) are correctly classified.