COMP 5790/ 6790/ 6796

Special Topics: Information Retrieval

Instructor: Shubhra ("Santu") Karmaker

Assignment #5: Mixture Language Models [100 points]

⚠ **Notice:** This assignment is due **Friday, February 21, 2020 at 11:59pm**.

Please submit your solutions via Canvas (https://auburn.instructure.com/). You should submit your assignment as a **typeset PDF**. Please do not include scanned or photographed equations as they are difficult for us to grade.

1. Mixture Models [25 pts]

Author *H* and author *T* are co-authoring a paper in the following way:

1. Each word is written independently.
2. When writing a word, they would first toss a coin to decide who will write the word. The coin is known to show up as "heads" 80% of the time. If the coin shows up as "heads", then author H would write the word, otherwise, author T would write the word.
3. If it is author H's turn to write, he would "write" the word by simply drawing a word according to word distribution $p(w|H)$. Similarly, if it is author T's turn to write, he would "write" the word by drawing a word according to word distribution $p(w|T)$.

Suppose the two distributions $p(w|H)$ and $p(w|T)$ are defined as follows:

Word w	$p(w\|H)$	$p(w\|T)$
the	0.3	0.3
computer	0.1	0.2
data	0.1	0.1
baseball	0.2	0.1
game	0.2	0.1
…	…	…

a. **[5 pts]** What is the probability that they would write "the" as the first word of the paper? Show your calculation.

b. **[5 pts]** What is the probability that they would write "the" as the second word of the paper? Show your calculation.

c. **[5 pts]** Suppose we observe that the first word they wrote is "data", what is the probability that this word was written by author H? Show your calculation.

d. **[5 pts]** Imagine that we observe a very long paper written by them (e.g., with more than 10,000 words). Among the 5 words shown in the table above (i.e., "the", "computer", "data", "baseball", "game"), which one would you expect to occur least frequently in the paper? Briefly explain why.

e. **[5 pts]** Suppose we don't know $p(w|H)$, but observed a paper D known to be written solely by author H. That is, the coin somehow always showed up as "heads" when they wrote the paper. Suppose D= "the computer data the computer game the computer data game" and we would like to use the maximum likelihood estimator to estimate $p(w|H)$. What would be the estimated probabilities of "computer" and "game", respectively? Show your calculations.

2. PLSA [75 points]

The PLSA algorithm models a corpus of documents $D=\{d_1,d_2,\ldots, d_N\}$ as a mixture of K different "topics" $\theta_1,\theta_2,\ldots,\theta_K$, each of which is a multinomial over words in a fixed vocabulary V. Each document d_i is associated with a distribution π_i which is a multinomial over the K topics and represents the likelihood that a word in that document is generated from a specific topic.

Under this model, we have a log likelihood of:

$$\log p(D \mid \Theta, \Pi) = \sum_{i=1}^{N} \sum_{j=1}^{|d_i|} \log \left\{ \sum_{k=1}^{K} p(z_{i,j} = k \mid \pi_i) p(d_{i,j} = w \mid \theta_k) \right\}$$

If we wished to find the maximum likelihood estimate of the parameters Θ and Π, we would have trouble coming up with an analytical solution due to the summation occurring inside of the logarithm. Thus, we turn to the EM algorithm to come up with a maximum likelihood estimator for the parameters instead.

In lecture, we discussed a modification of the original PLSA model that incorporated a fixed background topic. In that model, with probability λ, we generated a word from the background $p(w|D)$ and with probability $1-\lambda$, we generated it from a mixture of K different topics like the original PLSA model. Formally, we have a log likelihood of:

$$\log p(D \mid \Theta, \Pi) = \sum_{i=1}^{N} \sum_{j=1}^{|d_i|} \log \left\{ \lambda p(d_{i,j} = w \mid D) + (1-\lambda) \sum_{k=1}^{K} p(z_{i,j} = k \mid \pi_i) p(d_{i,j} = w \mid \theta_k) \right\}$$

i. PLSA With and Without the Background Topic [15 pts]

In this question, we will investigate the difference between these two models.

a. **[5 pts]** The second model (the one we covered in lecture) uses a fixed background topic with a fixed probability λ that is specified in advance. How can you configure this model so that it reduces to the first one? (In other words, for what setting of the parameters is the second model equivalent to the first?)

b. **[5 pts]** In the second model, $p(w|D)$ is our background language model which is assumed to be fixed throughout the parameter learning process. Give the maximum likelihood estimate for this language model.

c. **[5 pts]** Formulate a hypothesis (as in, a falsifiable statement) about the impact of λ on the topics $\{\theta_1,\ldots,\theta_K\}$ found by the model. What happens if λ is very large? What if it is very small?

Describe how you could test your hypothesis systematically. (In other words, describe an experiment that could prove or refute said hypothesis.)

ii. Deriving the EM Algorithm for PLSA with a Background Topic [15 pts]

a. **[8 pts]** In the discrete case, it is common to formulate EM algorithms as computing and normalizing expected counts of events. Let $n_{d,k}$ be the number of times we expect to see any word token in document d assigned to topic k, and let $n_{w,k}$ be the number of times we expect the word type w to be assigned the topic k.

Write down the equations for computing $n_{d,k}$ and $n_{w,k}$. Computing these across the entire corpus is your E-step calculation.

b. **[7 pts]** Write down the formula for re-estimating the parameters Θ and Π in terms of the expected counts you computed above. This is your M-step.

iii. Implementing PLSA with a Background Topic [45 pts]

a. **[20 pts]** Now, let's implement the PLSA model with the background topic. You may use any programming language you would like,

Implement the EM algorithm you derived above. Make sure that you are able to set a specific random seed for your random initialization (that is, the seed you use to initialize your random number generator that is used to create the initial random starting parameters $\Theta(0)$ and $\Pi(0)$).

You should terminate your algorithm after 100 iterations, or when the relative change in the log likelihood of the data is less than 0.0001.

The relative change in log likelihood is:

$$\Delta_i = \frac{\log p(D \mid \Theta^{(i-1)}, \Pi^{(i-1)}) - \log p(D \mid \Theta^{(i)}, \Pi^{(i)})}{\log p(D \mid \Theta^{(i-1)}, \Pi^{(i-1)})}$$

At the end of each iteration, print both the log likelihood and the relative change in log likelihood.

b. **[10 pts]** Now run your PLSA implementation on the dataset of DBLP abstracts we provided in Canvas files section with K=20 and λ=0.9.

Make both of the plots from above (log likelihood and relative difference in log likelihood) with the actual data now. Briefly explain why each is shaped the way that it is. Did this match your intuition?

c. **[15 pts]** Print out the top 10 words in each of the 20 topics you found above with λ=0.9. Now set λ=0.3 and print the top 10 words in each of the 20 topics.

Describe the difference between the topics you found with a large λ vs a small λ. Is this what you expected? Explain why you observe this phenomenon.

Based on this, if someone wanted to find topics that have a very descriptive top ten words, what would you recommend to them for setting λ?