Auburn University
Department of Computer Science and Software Engineering

COMP 5970/ 6970/ 6976– Midterm Exam #2
Information Retrieval: Spring 2020

Instructor: Dr. Shubhra (Santu) Karmaker

April 17, 2020

Name: _____

NetID: _____

This exam contains 13 pages (including this cover page) and 8 questions. Total of points is 100. Good luck!

Distribution of Marks

Question	Topic	Points	Score
1	Evaluation	10	
2	Language Models	10	
3	EM Algorithm	20	
4	Mixture Model	20	
5	Feedback	10	
6	Distributed Semantics	20	
7	Learning to Rank	5	
8	Recommender Systems	5	
Total		100	

1. **[10 points] Evaluation**

   Questions (a)-(f) refer to the following scenario of two systems:
   Suppose a query has a total of 4 relevant documents in a collection with 100 documents. System A and System B have each retrieved 10 documents, and the relevance status of the two ranked lists of results is:

   System A: [+,+,-,-,-,-,-,-,-,-]
   System B: [+,-,+,-,-,-,-,-,-,-]

   where a "+" (or "-") indicates that the corresponding document is relevant (or non-relevant). For example, the first document from System B is relevant, while the second is non-relevant, etc.

   (a) **[1/10 point]** What is the precision of System A?
   (A) 2/4 (B) 2/8 (C) 2/10 (D) 4/10

   (b) **[1/10 point]** What is the recall of System A?
   (A) 2/4 (B) 2/8 (C) 2/10 (D) 4/10

   (c) **[3/10 points]** What is the average precision of System A?
   (A) 2/4 (B) 2/8 (C) 2/10 (D) 4/10

   (d) **[1/10 points]** If we compare System A and System B based on precision at top five documents (i.e., prec@5doc), which system is better?
   (A) A is better than B (B) B is better than A (C) A and B have the same prec@5doc

   (e) **[1/10 points]** If we compare System A and System B based on average precision, which system is better?
   (A) A is better than B (B) B is better than A (C) A and B have the same average precision.

   (f) **[3/10 points]** Which of the following general statements about Mean Average Precision (MAP) and NDCG are true?

   (A) When binary relevance judgments are made, both MAP and NDCG can be used to measure ranking accuracy.
   (B) Because NDCG is normalized per query, it can better ensure comparability between different queries than MAP.
   (C) When more than two levels of relevance judgments are made, we cannot use MAP for evaluation.
   (D) Both NDCG and MAP are more sensitive to small differences in ranking than precision at k documents.

2. **[10 points] Language Models for Retrieval**

Let $q = q_1...q_m$ be a query, d be a document, $p(q_i|d)$ be the probability of query word q_i according to a smoothed document language model estimated based on d, and $p(w|C)$ be the collection (background) language model.

(a) **[3/10 points]** When refining the query likelihood scoring function, we often write down the following:

$$p(q|d) \overset{\text{rank}}{=} \sum_{i=1}^{m} \log p(q_i|d)$$

where $\overset{\text{rank}}{=}$ means the two sides of the equation are equivalent for ranking documents with respect to the same query. What assumption(s) have we made when we make the refinement shown above?

(b) **[7/10 points]** The equation above can be further written as:

$$p(q|d) \overset{\text{rank}}{=} \sum_{w \in V} c(w, q) \log p(w|d)$$

where w is a word in our vocabularly set V, and $c(w, q)$ is the count of word w in query q. This new formula can also be interpreted as a vector space model. If we make this interpretation, what would be the query vector? What would be the document vector? What would be the similarity function? If we use Jelinek-Mercer smoothing (i.e., fixed coefficient linear interpolation) with the smoothing parameter λ indicating the amount of smoothing, what would be the weight of term w in the document vector if $c(w, D)$ is the count of word w in document d, and the length of document d is $|d|$? Does the term weight in the document vector capture TF-IDF weighting and document length normalization heuristics? Why?

3. **[20 points] EM Algorithm**

Given two documents D_1 and D_2, and a background language model $p(w|C)$. We can use the maximum likelihood estimator to estimate a unigram language model based on D_1, which will be denoted by θ_1 (i.e., $p(w|\theta_1) = c(w, D_1)/|D_1|$)). Now, we can assume that document D_2 is generated by sampling words from a two-component multinomial mixture model where one component is $p(w|\theta_1)$, and the other is $p(w|C)$. Let λ denote the probability that $p(w|\theta_1)$ would be selected to generate a word in D_2 (thus, $1 - \lambda$ would be the probability of selecting the background model $p(w|C)$). Let $D_2 = w_1 w_2 ... w_k$ where $w_i \in V$ is a word in our vocabulary set V. We can then use the mixture model to fit D_2 and compute the maximum likelihood estimate of λ, which can then be used to measure the redundancy of D_2 with respect to D_1. We can use the EM algorithm to compute the maximum likelihood estimate.

(a) **[4/20 points]** Write down the formula to compute the probability of generating a word w_i in document D_2?

(b) **[2/20 points]** Write down the log-likelihood of the whole document D_2, i.e., the probability of observing all the words in D_2 being generated from the mixture model.

(c) **[6/20 points]** How many binary hidden variables in total do we need for computing this maximum likelihood estimate?

(d) **[8/20 points]** Write down the E-step and M-step formulas for estimating λ.

4. **[20 points] Mixture Models**

In this task, we want to estimate the language models of a two-component mixture model that we are assuming was used to generate a document $d = (w_1, w_2, ..., w_N)$. Specifically, we're given a mixing parameter λ that denotes the probability that the language model $p(w|\theta_0)$ would be selected to generate the word. With probability $1 - \lambda$, another language model $p(w|\theta_1)$ will be selected to generate the word. We can use the EM algorithm to compute the maximum likelihood estimate for θ_0 and θ_1. As usual, let $c(w, d)$ be the number of times we observe vocabulary word w occurring in the document d.

(a) **[4/20 points]** Write down the likelihood for the described two-component mixture model.

(i) $p(d|\theta_0, \theta_1, \lambda) =$

(ii) This expression is: **(A)** The complete likelihood **(B)** The incomplete likelihood

(b) **[4/20 points]** We can now introduce random variables (i.e., hidden variables) z_w that equal 0 if word w is drawn from the language model θ_0 and 1 if w is drawn from the language model θ_1 to generate the document d. Write down the likelihood including random variable $z = (z_{w_1}, z_{w_2}, ..., z_{w_M})$ where M is the size of the vocabulary. You may find it convenient to use the notation $a^x b^{1-x}$ to denote a function that has a value of a when the binary variable x = 1 and a value of b when x = 0.

(i) $p(d, z|\theta_0, \theta_1, \lambda) =$

(ii) This expression is: **(A)** The complete likelihood **(B)** The incomplete likelihood

(c) **[4/20 points]** Give the following formulas in terms of the parameters λ, $p(w|\theta_0)$ and $p(w|\theta_1)$

(i) $p(z_{w_i} = 0|w_i) =$

(ii) $p(z_{w_i} = 1|w_i) =$

(d) **[6/20 points]** Write down the maximization step of the EM algorithm for θ_0 and θ_1:

$p(w|\theta_0) =$

$p(w|\theta_1) =$

(e) **[1/20 points]** The expectation maximization (EM) algorithm cannot be guaranteed to converge to a global optimum unless there is only one maximum value (only one local maximum).

true **false**

(f) **[1/20 points]** Briefly describe one way of determining when to stop running iterations of the EM algorithm.

5. **[10 points] Feedback in Retrieval**

On the surface, the KL-divergence retrieval function is very similar to a vector space model retrieval function, but unlike dot product or cosine measure typically used in a vector space model, the KL-divergence function is asymmetric. That is, $D(\theta_Q||\theta_D) \neq D(\theta_D||\theta_Q)$. An interesting question is thus which way of using the KL-divergence function is more reasonable from the perspective of retrieval. $D(\theta_Q||\theta_D)$ can be heuristically justified based on the fact that it generalizes query likelihood retrieval function. What do you think about ranking documents based on the following alternative KL-divergence function (AKL)?

$$score(Q, D) = -\sum_{w \in V} p(w|\theta_D) \log \frac{p(w|\theta_D)}{p(w|\theta_Q)}.$$

where θ_D is a unigram document language model capturing the content of document D, and θ_Q is a unigram query language model capturing the information need expressed by query Q.

One way to estimate θ_D and θ_Q is to use the maximum likelihood estimator:

$$p(w|\theta_D) = \frac{c(w, D)}{|D|}$$

$$p(w|\theta_Q) = \frac{c(w, Q)}{|Q|}$$

(a) **[5/10 points]** What is a major problem with this way of estimating θ_Q from retrieval perspective? Can you propose a way to solve this problem?

(b) **[5/10 points]** After you fix this problem, analyze how well various retrieval heuristics (e.g., TF weighting, IDF weighting, and document length normalization) are implemented in this retrieval function and analyze whether this AKL retrieval function would work as well as the regular KL-divergence retrieval function.

6. **[20 points] Distributed Semantics**

   (a) **[5/20 points]** What is Distributional Hypothesis? How does CBOW and Skipgram capture this Hypothesis?

   (b) **[5/20 points]** Let $X(w)$ be the embedding vector for word w learned using a word embedding algorithm such as Glove or Word2Vec. Consider three vectors X("Apple"), X("Steve Jobs"), X("Microsoft"), which represent the three word vectors representing words "Apple", "Steve Jobs" and "Microsoft", respectively. How would you use these three vectors to answer the following analogical question?

   "Who is the founder of Microsoft?"

(c) **[5/20 points]** This questions refers to the paper by Tomas Mikolov et al. "*Distributed Representations of Words and Phrases and their Compositionality*". In: Advances in Neural Information Processing Systems 26.2013, pp. 3111–3119.

As a result of the Negative Sampling approach described in the paper for training word embedding, is it possible for a pair of word (a, b), where a is the target word and b is the context word, that, pair (a, b) is considered as both a positive and negative example during training? If possible, how? If not possible, why?

(d) **[5/20 points]** Both Skipgram and Glove paper replace the original (hard) learning problem into a much simpler and computationally feasible alternative machine learning problem. Which of the following alternative learning problem was formulated by Skipgram? What was done for the same in case of Glove? Justify your answer.

Clustering	Learning to Rank
Regression	Binary Classification
Multi-Class Classification	Forecasting a time series

7. **[5 points] Learning to Rank**

   Learning-to-rank (LETOR) methods can be categorized into the follow three types:

   - Pointwise LETOR method
   - Pairwise LETOR method
   - Listwise LETOR method

   Among these three methods, which one do you think is more appropriate for practical scenario? Why?

8. **[5 points] Recommender Systems**

   Collaborative filtering is essentially making filtering decisions for an individual user based on the judgments of other users. It heavily relies on the availability of large number of historical user preferences. However, when a Collaborative Filtering based Recommender Systems is deployed for the first time, such historical data is not immediately available. This is called a "cold start" problem.

   Describe one possible solution to the "cold start" problem.

This page is intentionally left blank to accommodate work that wouldn't fit elsewhere and/or scratch work.

This page is intentionally left blank to accommodate work that wouldn't fit elsewhere and/or scratch work.