Practice Questions for CS410 Midterm Exam

Please feel free to discuss these questions with your classmates

1. Let M be the unigram language model representing the “text mining topic” shown on slide 22 of the NLP lecture. If we draw two words independently according to this model, what is the probability of obtaining the word sequence “text mining”? What is the probability of obtaining “mining text”?
2. On slide 23 of the NLP lecture, according to the maximum likelihood estimate, what is the estimated probability p(“algorithm”|θ)? Suppose we duplicate the text document by concatenating it with itself to obtain a new document twice as long as the original one, and estimate the language model based on the new document. Would we obtain different estimated probabilities? What if we randomly delete a word from the document and then estimate the language model based on the new document?
3. Probability ranking principle states that ranking documents based on their probabilities of relevance to a query is optimal under two assumptions: (1) a user would sequentially browse documents in the presented order; (2) the usefulness of each document to a user is independent of the usefulness of other documents. Can you give an example where the second assumption is clearly not true in real application? That is, can you think of a scenario where the usefulness of one document to a user is clearly affected by usefulness of other documents that a user may have already seen? In such a case, ranking documents based on their individual scores would not be optimal. Can you think of a way to improve the results?
4. Let D be a document in a text collection. Suppose we add a copy of D to the collection (i.e., D is duplicated). How would this affect the IDF (Inverse Document Frequency) values of all the words in the collection? Why?
5. The BM25 retrieval function is of the following form:

   $$\text{Score(Q,D)}= \sum_{w \in Q,D} f1(c(w,Q))IDF(w)TF(c(w,D))$$

   Where c(w,Q) and c(w,D) are the count of word w in the query Q and document D, respectively. Compare this with the BM25 formula and figure out the exact form of the formula for the TF(c(w,D)) part. This part contains a parameter (b) to control document length normalization. What would happen if b is set to zero? Set to 1? For what kind of documents, does TF(c(w,D)) NOT depend on the document length no matter what value b is set to? Another parameter k1 controls the maximum possible value of TF(c(w,D)). Why do you think it might be beneficial to set an upper bound on TF(c(w,D))? What undesirable consequence would happen if we don’t set an upper bound?

6. According to Zipf's law, which of the following strategies is more effective for reducing the size of an inverted index? (1) reduce k common words; (2) remove k rare words.
7. Is it possible to have a gamma code with an even number of bits? Why? What number does the gamma code 111101101 encode? What's the gamma code for the number 23?
8. Suppose the relevance status of the top-8 ranked results from a system is [+,+,+,-,-,-,-,+]. Suppose there are in total 10 relevant documents in the collection. Compute the following evaluation measures for this result: (1) Precision. (2) Recall. (3) F1. (4) Precision at 5 documents. (5) Average Precision.
9. When we compute the average performance of a method over a set of queries, we may choose to either take an arithmetic mean, leading to Mean Average Precision (MAP), or take a geometric mean, leading to Geometric Mean Average Precision (gMAP). Why is it possible that system A outperforms system B in MAP, but B outperforms A in gMAP? In such a case, which one would you trust, MAP or gMAP?
10. Is it possible that one system outperforms the other in terms of MAP by loses to the other in terms of precision at 10 documents?
11. Compare NDCG with MAP and point out their similarities and differences.
12. Why is it necessary to do statistical significance test when comparing two retrieval systems in terms of their retrieval accuracy?
13. How does adding a word to the query affect the query likelihood value, p(Q|D)? Does this increase or decrease the likelihood of the query? Why?
14. Suppose a word w has occurred 10 times in a document D with 100 words. Assume that the probability of the word according to the collection (background) language model, p(w|C) is 0.01, and the Dirichlet prior smoothing parameter is 300. What is the estimated probability of this word in the document language model p(w|D) if we use Dirichlet prior smoothing? If we increase the smoothing parameter, would the estimated probability p(w|D) become larger or smaller? Why?
15. Compare the KL-divergence retrieval function with the query likelihood retrieval function. What's the similarity and what's the difference?
16. Compare Rocchio feedback with the two-component mixture model feedback method. In what sense are they similar?
17. In order to compute PageRank, what matrix needs to be constructed? What pages will have high values according to PageRank? Which of the following is most likely effective for increasing the PageRank score of a page: (1) adding an inlink; (2) adding an outlink; (3) deleting an inlink; and (4) deleting an outlink? Which is most likely going to decrease the PageRank score of the page?
18. How do you design a MapReduce program to generate counts of all the nouns in a collection? You can assume that you have available a part of speech tagger running on a Hadoop cluster.

19. Web search engines use a machine learning approach to combine multiple scoring factors (also called features). Can you give at least two examples of such scoring factors? That is, give at least two different ways to score/rank Web pages. These machine learning approaches need training data to help decide the parameter values that control how features are combined. How can they possibly obtain such a training data set?
20. What's the computational complexity of the memory-based collaborative filtering algorithm if we compute the prediction of rating for a (user-object) pair in a brute force way? Is it possible to leverage an indexing approach (e.g., inverted index) to potentially speed up the memory-based collaborative filtering algorithm?
21. Can you design an algorithm based on Rocchio to perform text categorization?
22. When using a retrieval toolkit to perform k-nearest neighbor (kNN) document classification, what is our "query"? Do you think the idea of pseudo feedback might also be applicable to document categorization to potentially improve classification accuracy?
23. Suppose we use the number of times a term occurs in all the documents to form a vector to represent each term. For example, if a term T occurred once in document 1, 10 times in document 3, 5 times in document 4, …, we would have a vector like V(T)=(1, 0, 10, 4,…). Suppose we use dot-product or cosine measure to compute the similarity between two vectors representing two terms. What kind of term pairs would have the highest similarity? Suppose we do clustering of terms based on such a similarity function on a collection of product reviews from Amazon. Can we expect to obtain some meaningful clusters of terms? For example, could we expect feature terms describing a particularly kind of products (e.g., cell phones) be grouped together? Why? If we hope to separate terms describing different products into different clusters, do you think adding IDF weighting to the weight of each element in a vector would be helpful?
24. Why is average link more robust than single link or complete link for clustering?
25. Do you think the problem of exploration-exploitation tradeoff also matters for a regular search engine like Google? If so, can you apply Maximal Marginal Relevance (MMR) reranking algorithm to make a search engine more exploratory?