IR Qualification Preparation

Frederic Go

1 Indexing

1. (28 pts) Inverted index can improve the speed of query processing in IR systems.
   i. (a) (6 pts) Give two advantages of controlled-vocabulary indexing over free-text indexing. **Ans.**1) we have clear ontology and relationship among words 2) no vocabulary mismatch (each word has clear meaning and we know what thing the speak addresses)
   ii. (b) (6 pts) What are term-at-a-time and document-at-a-time query processing methods, respectively? Give an advantage of term-at-a-time query processing over document-at-a-time query processing. /ans let query $q = q_1, ..., q_n$, q_i the i-th token. These two methods are the strategy of calculating all scores $s(d_i, q)$. Document-at-a-time: we load every postings for $\{q_1, .., q_n\}$. for each document d we sum scores $s(d, q_i)$ for all q_i and keep a priority queue on document scores. Term-at-a-time: for each token q_i, we only extract one posting, and increase the scores for each document see.but we need to have a accumelator for the scores of each documents. Advantages: only access one posting means we don't have to read out postings from disk many times.
   iii. (c) (8 pts) Give an example to describe how to effectively compress a postings list, and then discuss how document frequency of a word affects the compression ratio of its posting list. **Ans.**Delta coding: save first document's id, and the following documents the differences between two adjacent ids. Then encode will a coding which has length proportional to deltas. Ex: 1 - 123- 133 - 150 (coding)1- 122-10-17. The compression ratio depends on document frequency fo this word, because when document frequency is high, most numbers will be very small. when document frequency is low, numbers are large and the encoded posting is large.
   iv. (d) (8 pts) Write an algorithm to efficiently intersect two given postings lists with skip pointers, and then discuss how the number of skip pointers affects the complexity of your algorithm. **Ans.**p1; p2 intersect =[]; a = p1.first; b = p2.first while a != p1.end or b != p2.end: if $a < b$: if p1.pointer ¡ b: p1.jump a = p1.next
   if $a > b$: if p1.pointer ¡ b: p1.jump b = p2.next if a == b: intersect.append(a)
   Analysis: suppose list is n-byte, each skip pointer has k bytes, and the interval between skip pointers is c-byte. To find p documents in both list, we need to read on average $kn/c + pc/2$. Why? n/c reads of skip pointers and k for each skip pointer read, suppose we cannot match any documents. suppose we have to scan half of the c-byte interval, we have $pc/2$ to find all p documents.

2 Vector Space Model

1. (20 pts) Several variants of term-weighting for vector space model have been developed.

i. (6 pts) The logarithm function is often used for calculating some weights. Give one example formula for such weight. Explain the rationale behind the usage of logarithm as clearly as possible. **Ans.**For example, IDF (inverse document frequency) ,

$$IDF(t) = \log \frac{n}{k},$$

where n is the total number of documents and k is the number of documents containing term t. Because we think rare words, words occuring in fewer documents, are more important, we may consider $\frac{n}{k}$. But this kind heuristic is linear, meaning that the weights can be equally important for all values of $\frac{n}{k}$ up to infinity. Often, the linear assumption is wrong, and the effect of very high value may not be as strong as the low values. Therefore we introduce nonlinearity by using log to penalize the effect of high values.

ii. (10 pts) Here is the way to transform term frequency (TF) in Okapi BM25:

$$(k+1)\frac{\text{TF}}{k+\text{TF}} \text{ (k is a non-negative number)}$$

Whats the meaning of parameter k? Discuss the cases where k = 0 and k = . Whats the upper bound of the transformed TF? Draw a figure to show the relationship between original TF and transformed TF. **Ans.**k turns on/off the effect of term frequency. When $k = 0$ this score is always 0 or 1, thus the TF is binary. When $k > 0$, higher TF is penalized. When k is large, the transformed $TF = TF$.

iii. (4 pts) We want to rank documents d_i (i=1..N) w.r.t. query q. Prove that if q and d_i are all normalized to unit vectors, then the rank ordering produced by Euclidean distance is identical to that produced by cosine similarities. **Ans.**
Cosine similarity:$cosine(\vec{d}, \vec{q}) = \sum_i d_i q_i$ (because q and d are normalized)
Euclidean distance: $dist(\vec{d}, \vec{q}) = \sqrt{\sum_i (d_i - q_i)^2} = \sqrt{\sum_i (x_i^2 + q_i^2 - 2x_i q_i)} = \sqrt{2 - 2\sum_i x_i q_i}$,
similarity is proportional to $\sum_i x_i q_i$

3 Probabilistic Model

1. (20 pts) The basis of probability ranking principle (PRP) indicates that a retrieval system performs optimally if documents are ranked according to decreasing probabilities of their relevance to a query. The binary independence model (BIM) is developed based on PRP.

   i. (a) (8 pts) Describe how BIM with relevance feedback estimates the probability that a word appears in a document relevant or irrelevant to a query. **Ans.**Denote the probabitlity that a word presents in relevant document $p_i = P(x_i = 1|R, q)$, and $u_i = P(x_i = 1|NR, q)$. We can initially guess that $p_i = 0.5$ (random number), and $u_i = df_i/N$, where df_i is x_i's document frequency and N total number of document, reasonable because most documents are assume to be nonrelevant. After user has picked up some documents in the highest ranked list, say top k documents, we can partition this list as $V = VR \cap VNR$, where VR denotes the set of relevant documents and VNR denotes the set of nonrelevant documents. Then we calculate $p_i = \frac{|VR_i|}{|VR|}$, with VR_i the set of relevant document where x_i is present, and $u_i = \frac{|VNR_i|}{|VNR|}$. Then we iterate these two process until convergence.

   ii. (6 pts) Give an example to explain that ranking based on PRP may not provide satisfactory search results even if the probability of relevance can be well estimated. **Ans.**PRP's most

annoying feature is that it judges the relevance of documents independently. If this is true, suppose that on the web there many highly similar documents and these documents are highly relevant to query x. At the same time there are some other less relevant documents discussing other interesting topics. According to PRP, the top of list will be occupied by the former group of documents and annoy the user.

iii. (6 pts) Give two advantages of TF-IDF-based vector space model over BIM. **Ans.**TFIDF: consider word frequency BIM for short documents. TFIDF: doesn't need relevance feedback to estimate parameters. TFIDF: can consider document length effects.

4 Language Model

1. 4. (20 pts) Statistical language models (LM) have been successfully applied to IR.
   i. (a) (10 pts) What are query likelihood LM and document likelihood LM, respectively? Give an advantage of query likelihood LM over document likelihood LM in IR. **Ans.**Given a document d and query q, we want the probability of relevance $P(R|d,q)$. With Bayes theorem, we have $P(R|d,q) \propto P(q|d,R)P(d|R)$. Suppose the prior $P(d|R)$ is uniform, what we have left is calculating $P(q|d,R)$. This is called query likelihood, because quries are thought to be generated by document language models and the ranking are determined by how likely the query is genereated by these documents. If we use Bayes theorem in the other way, $P(R|d,q) \propto P(d|q,R)P(q|R)$. Assuming $P(q|R)$ to be uniform, then we will need to calculate $P(d|q,R)$. This time we need document generation model given a query. This is called document Languge model. Query likelihood LM is better to estimate because when estimating models we need parameters θ_d for $P(w_i|\theta_d)$ but for document likelihood LM we need to estimate θ_q for $P(w_i|\theta_q)$. The former models we have a lot of documents which are long and have many words, but in the latter models we only have queries which are short and often don't have a real language model.
   ii. (b) (10 pts) Propose a way to perform smoothing for a LM (show your calculation), and then discuss how the smoothing affects retrieval performance for (1) the query terms not occurring in a document and (2) the query terms occurring in a document. **Ans.**The easiest one is introduce an big corpus, i.e.

$$P(w_i|d) = \begin{cases} P_{seen}(w_i|C), & \text{if } w_i \in d \\ \alpha_d P(w_i|C), & \text{otherwise.} \end{cases} \tag{1}$$

   If query term is absent from document, the smoothing bypass the probability to the corpus probability. So the retrieval performace depends on the similarity between the document and language model of the corpus. If a term is present in the document, the probability is estimate only by the document.

2. Smoothing is very import to retrieval model.
   i. Describe (1) how to apply language model to the ranking problem (2) how to perform smoothing in the language model
   ii. Describe (1) how to apply LSI(latent semantic indexing) to ranking problem and (2) how to perform low-rank approximation, i.e. smoothing, by dimension reduction. **Ans.**Denote term-to-document matrix A. Apply SVD to A, we have $A = U\Sigma V^\top$. U is term-to-concept matrix, Σ is the singular value matrix which stores the importance of each concept

dimension, and V is document-concept matrix. Now we apply LSI to ranking problem. Given a query, this can be represented as a document vector q. We use $\Sigma^{-1}U^\top$ to map q to the concept space. why? $A = U\Sigma V^\top$, therefore $\Sigma^{-1}U^\top A$ is concept-to-document matrix. then we calculate similarity between q and documents d_i in the collection, i.e. $sim(q, d_i)$ (2) dimension reduction is easy. since Σ is very large matrix $|terms| * |documents|$. we can delete smaller singular values and reduce this to a smaller concept space. This has an effect of smoothing because the reduced decomposition no longer fit to the original matrix A, but many zeroes are filled.

5 Evaluation

This kind of question is very popular because one can calculate the numbers.

1. Given a query q, there are 6 relevant document in a 1000-document colloection. Each system in two returns its top 10 list:

   System 1: R N R N N N N N R R

   System 2: N R N N R R R N N N

   i. Calculate MAP for the two systems. Which is higher? **Ans.**System 1: $\frac{1}{4}(1+\frac{2}{3}+\frac{3}{9}+\frac{4}{10}) = 0.6$, System 2: $\frac{1}{4}(\frac{1}{2}+\frac{2}{5}+\frac{3}{6}+\frac{4}{7}) \approx 0.5$. System 1 has higher MAP.

   ii. Intuition make sense? What does it mean to get high MAP? **Ans.**Ranking matters. Precision is discounted if a document ranks lower in the list, therefore system 2 has lower MAP.

   Now consider System 1. Suppose system 1 now returns 1000 document in a ranked list. And these are the top 10.

   iii. What are the largest and smallest possible MAP values, respectively, that system 1 could have? **Ans.**Because we have only seen 4 of the relevant documents, there are 2 left in the list. Therefore, the largest possible answer is $\frac{1}{6}(1+\frac{2}{3}+\frac{3}{9}+\frac{4}{10}+\frac{5}{11}+\frac{6}{12})$ and the smallest possible MAP is $\frac{1}{6}(1+\frac{2}{3}+\frac{3}{9}+\frac{4}{10}+\frac{5}{999}+\frac{6}{1000})$.

   iv. How larget the error can be if we use top 10 to estimate MAP instead of full list of 1000? **Ans.**If all relevant in top 10. estimation error = 0. The largest error case that all relevant documents falls into 11-16, in which case $MAP_{10} = 0$ but $MAP_{1000} = \frac{1}{6}(\frac{1}{11}+\frac{2}{12}+\frac{3}{13}+\frac{4}{14}+\frac{5}{15}+\frac{6}{16})$.

   v. Evaluation measure appropriate for question answering system. **Ans.**Mean reciprocal ranking, $MRR = \frac{1}{|Q|}\sum_Q \frac{1}{rank_i}$. Because question answering system only return 1 answer, so the precisions of highest ranked document for queries are averaged. Precision for the highest ranked is called reciprocal rank.

2. (16 pts) NDCG (Normalized Discounted Cumulative Gain), MAP (Mean Average Precision), Precision@k and F-measure are famous evaluation metrics in IR.

   i. (6 pts) What are the important aspects of relevance that are considered by NDCG but not by F-measure? **Ans.**F-measure does not consider the effect of ranking. It is in some sense and average of precision and recall. Precision and recall consider only overall ratios but cfNDCG is sensitive to the ordering of documents.

ii. (6 pts) Compare the performance of two IR systems. Give an example to explain if its possible that one performs better in terms of MAP but worse in terms of Precision@k? Show your calculation.

iii. (4 pts) Consider the balanced F-measure. What is the advantage of using the harmonic mean rather than arithmetic or geometric mean in its formula?

PageRank, HITS, preprocessing...

6 Miscelly

1. Most of search engines extract two kinds of features from web pages and structures, including topical features (e.g., terms) and quality features (e.g., PageRank), and apply machine learning approaches to learning the ranking.

   i. (8 pts) Describe four lexical processing methods for topical features. **Ans.**Tokenization: convert stentences to tokens, basic units of vocabulary, phase recognition: use classifier to detect multi word tokens, Normalizaton: convert different form of the same entity to the same form, e.g. U.S.A., USA, America, stopword removal: remove most frequent words from documents because they carry less meaning, e.g. "it", "is", "I", stemming: convert morphological variants into the same equivalence class, e.g.: "dog", "dogs", "doggy" = "dog".

   ii. (9 pts) Suppose a transition probability matrix defines a Markov chain. Under what conditions will the PageRank scores reach a stationary distribution? **Ans.**the transition matrix A must be 1) stochastic $\sum_{j=0}^{n} A_{ij} = 1$, 2) irreducible: for every pair of node (a, b) , there is a path between a and b, 3) aperiodic , a state i is periodic if $\exists k > 1$, for all path from i and back to i has length multiple of k.

   iii. (6 pts) Supervised learning requires training data to learn a model. How do search engines obtain such training data in an automatic way? Give an example to demonstrate how to learn a model for determining the parameters that combine topical and quality features. **Ans.**Search engine uses clickthrough data. When a user click a link, we have a positive sample.