University of Virginia
Department of Computer Science

CS 4501: Information Retrieval
Fall 2015

2:00pm-3:30pm, Tuesday, December 15th

Name:
ComputingID:

- This is a **closed book** and **closed notes** exam. No electronic aids or cheat sheets are allowed.
- There are 9 pages, 4 parts of questions (the last part is bonus questions), and 110 total points in this exam.
- The questions are printed on **both** sides!
- Please carefully read the instructions and questions before you answer them.
- Please pay special attention on your handwriting; if the answers are not recognizable by the instructor, the grading might be inaccurate (***NO*** argument about this after the grading is done).
- Try to keep your answers as concise as possible; grading is *NOT* by keyword matching.

Total	/110

Academic Integrity Agreement

I, the undersigned, have neither witnessed nor received any external help while taking this exam. I understand that doing so (and not reporting) is a violation of the University's academic integrity policies, and may result in academic sanctions.

Signature: _____

Your exam will not be graded unless the above agreement is signed.

1 True/False Questions (12×2 pts)

Please choose either True or False for each of the following statements. For the statement you believe it is False, please give your brief explanation of it (you do not need to explain when you believe it is True). Two point for each question. *Note: the credit can only be granted if your explanation for the false statement is correct.*

1. Depth-first crawling is preferred in vertical search engines.
   ***False**, and Explain*: focused crawling should be applied, since it is not necessary the links will only point to the pages of the same topical domain.

2. Smoothing of a language model is not needed if all the query terms occur in a document.

   ***False**, and Explain*: smoothing has to be applied to all the documents during search, otherwise the query likelihood will not be comparable. Or you can argue unless all the query terms occur in all the documents, otherwise we still need smoothing.

3. Pairwise learning to rank algorithms are preferred over pointwise algorithms because they can directly optimize ranking-related metrics, e.g., MAP or NDCG.
   ***False**, and Explain*: pairwise learning to rank algorithms can only minimize the number of misorder pairs, but this cannot directly optimize ranking-related metrics.

4. In query generation models, we have assumed words in documents are independent.
   ***False**, and Explain*: we do not have such a assumption in query generation model at all, e.g., we can use N-gram language models.

5. Relevant feedback helps a document generation model to improve its estimation for current query and future document.
   ***False**, and Explain*: it helps a document generation model to improve its estimation for future query and current document.

6. PageRank score can be computed at the crawling stage.
   True
   ***False**, and Explain*: we need to have sufficient observations of the link graph before we can compute the PageRank.

7. Interleaved test is more sensitive than A/B test.
   True

8. Maximum a posterior estimation method for parameter estimation is problematic when one has only a small amount of observations.
   ***False**, and Explain*: Maximum a posterior estimation method is preferred when one only has a handful observations.

9. Three important heuristics in vector space ranking models are: term frequency, document frequency and query term frequency.
   ***False**, and Explain*: it should be term frequency, document frequency and document length normalization.

10. Because of position bias, result clicks can only be treated as implicit feedback.
***False**, and Explain*: because we do not know the exact reason why user clicks on the results, we can only treat click as implicit feedback.

11. The current way of human annotation based IR evaluation is biased, since we cannot exhaust the annotation of all relevant documents.
***False**, and Explain*: it can largely reveal the relative comparison between ranking algorithms.

12. The initial state of random walk in PageRank algorithm determines the estimation quality of its stationary distribution.
***False**, and Explain*: the stationary distribution of PageRank scores is only related to its transition matrix but independent of initial state of random walk.

2 Short Answer Questions (32 pts)

Most of the following questions can be answered by one or two sentences. Please make your answer concise and to the point.

1. For the queries below, can we still run through the inverted index join in time $O(m+n)$, where m and n are the length of the postings lists for *information* and *retrieval*? If not, what can we achieve? (4pts)

   - *information* **AND NOT** *retrieval*
     Yes, this is the difference between the posting lists of *information* and *retrieval*.
   - *information* **OR NOT** *retrieval*
     No, this essentially has to go over all the documents in the index, i.e., posting lists of *information* and all the other words except *retrieval*.

2. How does Zipf's law ensure effective inverted index compression? (4pts)
   By storing the gap between consecutive document indices, Zipf's law ensures that: 1) for frequent words, there will be many duplicated small gaps; 2) for infrequent words, the posting list is short. Therefore, by encoding the gap with variable codes, redundancy can be exploited for effective compression.

3. List three different types of click heuristics that generate pairwise result preference assertions. With the given click log of a user's search query, write out a pairwise assertion after each one of those heuristics. (6pts)
   Search log: $\{d_1^{(3)}, d_2, d_3^{(1)}, d_4^{(2)}, d_5, d_6^{(4)}, d_7, d_8, d_9, d_{10}\}$, where the subscript indicates document id, and $(\cdot)$ in the superscript represents click order.

   - Click > Skip Above, $d_4^{(2)} > d_2$
   - Last Click > Skip Above, $d_6^{(4)} > d_2$
   - Click > Skip Previous, $d_3^{(1)} > d_2$

4. List three different components/algorithms in a typical information retrieval system where sequential browsing has been assumed. (6pts)

   - Search result ranking, i.e., rank document by descending order of probability of being relevant
   - Search evaluation, i.e., higher weights are given documents ranked at higher position
   - Click modeling, examination event is assumed to be sequential from top to bottom

5. List three different behavior-based ranking metrics, that measure a retrieval system's search result quality based on users' interactive behaviors. Please also indicate their correlation with the search quality, e.g., an increased value corresponds improve ranking performance, or other way round. (6pts)

   - abandonment rate, increased value indicates worse ranking performance

- reformulation rate, increased value indicates worse ranking performance
- click per query rate, decreased value indicates worse ranking performance

6. Fill in the blanks to finish the derivation of ranking formula for the query generation model. (6pts)

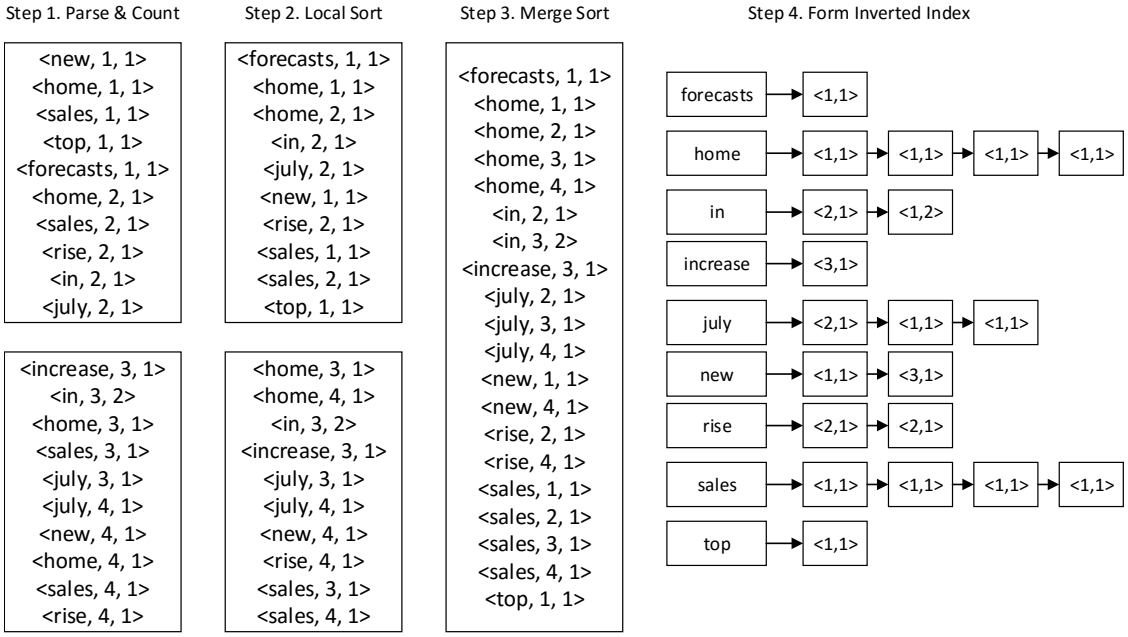
$$
\begin{aligned}
\log p(q|d) &= \sum_{w \in q} c(w,q) \log p(w|d) \\
&= \sum_{w \in q \cap d} c(w,q) \log p_{\text{seen}}(w|d) + \underline{\sum_{w \in q, w \notin d} c(w,q) \log \alpha_d p(w|C)} \\
&= \sum_{w \in q \cap d} c(w,q) \log p_{\text{seen}}(w|d) + \underline{\sum_{w \in q} c(w,q) \log \alpha_d p(w|C)} \\
&\quad - \sum_{w \in q \cap d} c(w,q) \log \alpha_d p(w|C) \\
&= \sum_{w \in q \cap d} c(w,q) \log \frac{p_{\text{seen}}(w|d)}{\alpha_d p(w|C)} + |q| \log \underline{\alpha_d} + \sum_{w \in q} c(w,q) \log p(w|C)
\end{aligned}
$$

3 Essay Questions (44 pts)

All the following questions focus on system/algorithm design. Please think about all the methods and concepts we have discussed in class (including those from the students' paper presentations) and try to give your best designs in terms of feasibility, comprehensiveness and novelty. When necessary, you can draw diagrams or write pseudo codes to illustrate your idea.

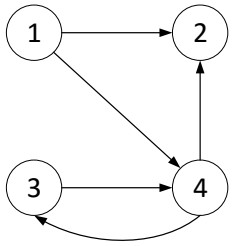
1. Given the following four documents in our archive, where the first two documents are stored in machine one, and the second two documents are stored in machine two, draw the procedure of MapReduce-based inverted index construction, and the resulting inverted index. (16 pts)

   - new home sales top forecasts
   - home sales rise in july
   - increase in home sales in july
   - july new home sales rise



The tuples in Step 1 to 3 are in the format of <term, docID, count>. And the tuples in the posting list of constructed inverted index are in the format of <gap, count>. The first tuple in the posting list is in the format of <docID, count>.
We will not give you penalty if your result inverted index stores document ID rather than the gap between document IDs.

2. Given the following hyperlink structure among four web pages, demonstrate the step by step construction of the transition matrix for PageRank with dumping factor $\alpha = 0.1$. You are suggested to write down the intermediate results. (12 pts)



1. adjacency matrix:

$$\begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

2. enable random jump on dead end:

$$\begin{bmatrix} 0 & 1 & 0 & 1 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

3. normalization:

$$\begin{bmatrix} 0 & 0.5 & 0 & 0.5 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0 & 0 & 0 & 1 \\ 0 & 0.5 & 0.5 & 0 \end{bmatrix}$$

4. enable random jump on all nodes:

$$0.1 * \begin{bmatrix} 0 & 0.5 & 0 & 0.5 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0 & 0 & 0 & 1 \\ 0 & 0.5 & 0.5 & 0 \end{bmatrix} + 0.9 * \begin{bmatrix} 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \end{bmatrix}$$

3. Bing wants to become a better personalized search engine. Combining the concepts and techniques that we have learned in this semester, please give your concrete suggestions of where and how an IR system can be personalized. Please cover at least three components in a typical retrieval system, e.g., query processing module, ranking functions, and feedback modeling. (16 pts)

   - In query processing module, we can add personalized query expansion or query topic classification based on users' query history. E.g., if a user often searches for programming languages, when he/she issues query "java", we can automatically append keywords like "programming language" to it (with relatively lower weight) and associate it with topical category "information technology", "programming language".
   - In ranking function module, we can build a personalized user profile based on Rocchio feedback or query language model based feedback techniques. In query time, we can rerank the initial search results by such user profile. Building different learning to rank algorithms to get personalized ranking feature combination is also feasible.
   - In feedback module, based different users' click behavior, we can build different examination models, e.g., some users tend to exam more results before click, while some users tend to open more links and read them one by one. These patterns indicate different relevance quality of the clicked results.
   - In result display module, we can also adapt to individual users' preference. For example, some users might prefer image vertical being displayed before the video vertical, some users might prefer read more text content. Such display settings can also be personalized accordingly.

4 Bonus Questions (10 pts)

All these questions are supposed to be open research questions. Your answers have to be very specific to convince the instructor that you deserve the bonus (generally mention some broad concepts will not count).

1. How to perform better mobile search? Credit is given based on the novelty of your proposal. (10pts)
   This is an open research question, and I am expecting some new ideas that are beyond what we have discussed in class, e.g., using locations and contacts.