CMPSCI 646, Information Retrieval **Fall 2014**
Final exam

This is a take-home final exam. It includes six questions (in several parts) worth a total of 100 points. You have 48 hours to complete it, starting from the moment you picked up this exam or that it was e-mailed to you (which might not be the same as the time the message arrived in your inbox).

Please submit your solutions using Moodle. A new topic has been created near the top called "Final Exam" that includes a link to an upload page. If you cannot upload your answers, then you may email your solutions to the professor or drop them off with him, with Ethem (the TA), or with Kate Moruzzi during normal business hours. You may also hand them in to the main office (ask them to time stamp the exam). If you need to fax your answers in, send them to 413-545-1789. Faxing is not recommended since readability might be affected.

The exam is open-book. You may use class notes, textbooks, papers, or other written resources. You may browse the internet. However, remember that if you use any source other than the class notes or discussion with the professor or TA, you ***must*** *cite your source. That is true whether you quote something or just use it to gain a better understanding of the material. If you quote something, you must indicate precisely where the quotation came from (including the page number). If by some unfortunate chance you find one of these questions answered exactly, remember to cite your source in that case, too. And, by the way, be cautious about trusting stuff that you find elsewhere; just because someone made it available on the internet does not mean that it's true.*

Finally, it is unacceptable for you to share this exam, the topics of the exam, or your answers with anyone else—regardless of whether they are in this class—until all exams have been returned, meaning until the 13th at 10:00am (the last possible end of a 48-hour window with an hour of slack time just in case). Sharing the exam or receiving a shared exam constitutes academic dishonesty and will result in a zero on the exam, an F in the class, or dismissal from the University. If someone attempts to share an exam with you, tell the professor immediately.

In general, questions worth few points should have short answers. Please keep that in mind when answering questions.

If you have questions or need clarification, please contact the professor or TA directly outside of Moodle. Do not post anything related to the exam to a Moodle forum.

See http://cs.umass.edu/~allan/FunAndGames for errata and clarifications.

Good luck.

Problem A. (18 points)

Problem C on the second midterm seemed to cause a great deal of confusion, so we are going to revisit it here. This question is about "pooling" and how it relates to evaluation. The setup for the problem is here, stated much more formally than in the midterm:

> Let *S* be a set of retrieval systems being evaluated. Every system $s \in S$ has produced ranked lists for the same queries that we will evaluate to determine which systems are better and which are worse. To create the relevance judgments, a set of systems $P \subsetneq S$ are selected and the top *n* ranked documents from each system in *P* are judged for relevance. When done, the collection of documents, *C*, consists of non-overlapping subsets, C_R of documents that were judged relevant, C_{NR} of documents that were judged non-relevant, and C_U of unjudged documents. By construction, any document $c \in C_U$ cannot have been in the top *n* documents of any system in *P* and a document $c \in C_R \cup C_{NR}$ must have been in the top *n* documents of at least one system in P, though it obviously could have appeared anywhere in the rankings of other systems.
>
> For purposes of evaluation, we assume that any document $c \in C_U$ is non-relevant.

1. Because the documents in C_U were never judged, we can be fairly certain that there are some relevant documents in the set. How should systems from *S* be selected to form pooled systems, *P*, that minimize the number of relevant documents in C_U?
2. Suppose that some document $x \in C_U$ is later judged and found to be relevant to a particular query *Q*. Suppose that when assuming that *x* was non-relevant, a system $s \in P$ has an average precision value of *AP* for that query and we know there are *r* relevant documents for *Q*. What is the greatest possible change in average precision if we recalculate it including knowledge of *x*'s relevance? You may assume that *n=100*.
3. In the same situation as (2), when would the *least* change in average precision occur?
4. Consider two systems, $p \in P$ and $t \in S \setminus P$. That is, *p* is a system that participated in the pooling (so its top *n* documents were judged) and *t* did not participate in the pooling, so there is no guarantee that its top *n* documents were all judged. Assume that to start with there are three known relevant documents for some query *Q* and they are at ranks 1, 2, and 3 for *p* and at ranks 2, 3, and 4 for *t*. Also assume that the document at rank 1 for *t* is in C_{NR} so we know it is non-relevant. Show that AP of *p* is greater than AP of *t*.
5. In the situation of (4), suppose that document *x* is now found as in (2). *Prove* whether knowledge about *x* changes or does not change your knowledge about the ranking of *t* and *p* by AP. Assume that *n=100*.
6. What does your answer to (5) suggest about the importance of obtaining judgments for systems outside of the initial pool, *P*? Support your answer.

Problem B. (12 points)

Briefly discuss issues involved in converting an IR system for news stories to a system for microblog entries (e.g., tweets). In the unlikely event you do not know what a "tweet" is, see the Wikipedia page on Twitter. Only describe changes, not things that remain the same. Does how or when indexing is done change? Should retrieval be handled differently? Should term weighting be done differently? What else? Support your answers and explain how you would evaluate your system to make choices.

Problem C. Presentation-based questions (5 points)

There were a good number of questions sent in. I selected a handful for this exam that I thought hit on topics that you should take away from the class.

1. Briefly explain why learning to rank does not directly optimize IR evaluation metrics.
2. Why is local context analysis (LCA) more successful than just expanding a query with the most frequent terms in the top-ranked documents?
3. True / False. The weighted sequential dependence model assigns importance weights to bigrams but uses the original query weight for unigrams.
4. Briefly explain why longer, more descriptive queries tend to underperform short "title" queries even though they contain more information.
5. One of the presentations discussed an approach where all words in documents and queries were converted to a representative of their synonym class (in this case, to the synonym in alphabetical order). What happens to the IDF values of terms in this situation?

Problem D. Short answer (15 points)

1. Suppose that you modified a query expansion algorithm (e.g., PRF or RM) so that only words that started with the same few characters (e.g., *run** or *info**) were included in the expansion. How would that compare to stemming?
2. Clustering by k nearest neighbors is not symmetric. That is, just because document D_1 is in D_2's cluster does not mean that D_2 will be in D_1's cluster. Show by example why not.
3. If a query has one relevant document, explain the difference between its average precision and reciprocal rank (as in MRR) scores.
4. If you heard the criticism that a signature file implementation increases the miss rate by 20%, what would be your reaction? Is that reasonable? Why or why not?
5. Suppose that four real IR systems – Indri, Galago, Lucene, and Terrier – are each set to use the same retrieval model: query-likelihood model with linear smoothing. You have managed to arrange things such that the parameters and settings are the same for all four systems such as stemmers and so on. We index a collection using these four IR systems and run 10 queries, each of which has 50 relevant documents. Which of these best describes what you would expect to see (and briefly justify your answer):
    a. The results of each IR system would be largely the same.
    b. The results of each IR system would be completely different.
    c. The results of Indri and Galago would be same; however, Lucene and Terrier would be different since Indri and Galago are open source and mostly for academic use but the others are not.

Problem E. Which is the winner? (18 points)

Pick either the vector space or language modeling approaches to information retrieval. In at most 250 words, argue that your choice is better than the other. (Hint: there is no right choice, so you should have no problems with either argument.)

Problem F. (22 points)

In another galaxy Gogolo is the most famous search engine and the engineers at Gogolo are about to move to use "learning to rank" in their system. Before doing that, they would like to perform toy experiments on a small dataset. They ask your consultancy since you are from a different galaxy and know more than they do.

First, they have extracted a number of features from their web documents and are not sure whether or not the features are useful. Please consider each feature below and indicate whether it is useful or not and (briefly) justify your answer.

1. PageRank, with the particular note that it is calculated independent of the query.
2. HITS and variations of PageRank that are *dependent* on the query (top ranked pages).
3. Anchor text—i.e., `<a href="URL">this is anchor text for URL</a>`
4. The fraction of terms in the stopword list that appear on the page (i.e., recall of all stopwords).
5. Click-through counts. That is, given a query, what URLs are clicked on across all users and how often?
6. Update recency. How recently was this page created or modified?
7. Metadata keywords provided by the author of the page.
8. The ratio of stop words to non-stop words in the page.
9. The depth of the URL path (number of slashes in the URL).

An important issue in "learning to rank" is normalization, or modifying features so that they have values in an expected range. Their goal is to normalize all feature values to be in the [0,1] range. Here is a sample of their data:

<Q1, D1>: 1:1 2:2 3:3 4:4 5:5
<Q1, D2>: 1:5 2:4 3:3 4:4 5:5
<Q2, D1>: 1:1 2:1 3:1 4:1 5:5
<Q2, D2>: 1:5 2:2 3:2 4:2 5:2

where each query/document pair has five features (1-5) with values. To normalize the features, the Gologo engineers will find a minimum and a maximum and then normalize a feature value as follows: $normValue = (value - minimum)/(maximum - minimum)$. They are considering three normalization schemes and would like your advice about which to use and, of course, your explanation of why it is the best choice.

a. Row-vector based normalization. In this case, each row is normalized by finding the max and min of the row and normalizing the values in that row.
b. Column-vector based normalization. Here, each column is normalized by finding the max and min of the column (feature number) and normalizing the values in that column.
c. Matrix-based normalization. For this situation, the max and min feature values are found across all rows and columns and then all values are normalized using those.

10. *(4 points)* Which normalization scheme should be used and why?