

# CMPSCI 646, Information Retrieval (Fall 2003)

## Midterm exam

Exam received:

*Due: Three days (72 hours) from when you receive the exam.* Please hand your exam into the instructor's mailbox, asking a secretary to write on the exam the time at which it was handed in. If you want to hand in the exam on Saturday or Sunday, you have two choices:

- Email an electronic copy of the exam to the instructor before the deadline, then provide a printed copy to Kate Moruzzi (room 368) on Monday.
- Contact Toni Rath (trath@cs.umass.edu) and/or Victor Lavrenko (lavrenko@cs.umass.edu) to arrange to drop off the exam with them. One or the other is likely to be in the office on the weekend and can note the time it was received. Please make this arrangement immediately.

If none of those options works for you, please contact the instructor immediately so we can attempt to make this work.

Your solution should be typewritten; if it is handwritten, it must be very neat (and will be difficult to hand in electronically if you have a Saturday or Sunday deadline).

**The work you hand in must be your own.** *You are not permitted to discuss or collaborate on problems with your classmates or with anyone else. You can obviously ask for help from the TA or the instructor.* When you hand in your exam, you may not discuss the exam with anyone else—regardless of whether or not they are done—until *after* Monday, November 17th, at 4:30pm. And hide your copy of the exam.

On the front of your exam solutions, please include the following statement and sign it: "I received no unauthorized help on this exam. The following is my own work."

This version of the exam is dated November 10, 1:45pm. Check the Web page for updates regularly.

This exam does not depend on any material after that on relevance feedback (Lectures 17&18). It includes four problems, each worth 25 points. Be sure you have all three pages.

### Problem CO (compression)

1. The problem of text classification can be described as follows. Given a set of classes,  $C = \{C_i\}$ , where each class contains documents, assign a new document  $D$  to the class  $C_i$  that is the best match for it. Generally "best match" means that they discuss the same topic or are otherwise very strongly related content. You can assume that each class has numerous training instances,  $D_{i,j} \in C_i$  for  $j = 1, \dots, |C_i|$ .

One approach to classification first builds a compression model for each class. When a new document,  $D$ , is considered for classification, it is compressed with respect to *each* class and is assigned to the one that yields the most compression. Why does that work?

2. Inverted lists are normally sorted in ascending order of document identifier. There are some situations where it instead makes sense to store them sorted by the descending weight of the term in the document—i.e., the first document listed will have the highest weight for that term, the second will have the next highest, and so on. What, if any, are the implications for inverted list compression of using that scheme instead? Support your claim.
3. LSI can be seen as a compression scheme because it reduces the number of dimensions in the vector space. Is LSI a "lossy" or a "lossless" compression scheme? Explain what it means to be whichever you choose, and support your claim.

## Problem LM (language models)

For these problems it will be helpful to have a copy of Ponte and Croft's SIGIR 1998 paper:

PDF <http://ciir.cs.umass.edu/pubfiles/ir-120.pdf>  
PostScript <http://ciir.cs.umass.edu/pubfiles/ir-120.ps>

Note that because of some font issues, the PDF version is not very legible on the screen, but it should print fine.

1. Is the Ponte and Croft a query likelihood model, document likelihood model, a combination, or something else?
2. The probability estimator they provide is,

$$\hat{p}(t|M_d) = \begin{cases} p_{mt}(t, d)^{(1.0-\hat{R}_{t,d})} \times p_{avg}(t)^{\hat{R}_{t,d}} & \text{if } tf_{t,d} > 0 \\ \frac{c_{f_t}}{cs} & \text{otherwise} \end{cases} \quad (1)$$

Although that is a fine probability estimator for the Ponte and Croft model (a Bernoulli model), it is not appropriate for the multinomial model that dominates language modeling in IR. Why isn't  $\hat{p}$  appropriate then?

3. Regardless of how the probabilities are estimated, the final function that is calculated for each document is,

$$\hat{p}(Q|M_d) = \prod_{t \in Q} \hat{p}(t|M_d) \times \prod_{t \notin Q} (1.0 - \hat{p}(t|M_d)) \quad (2)$$

This is a multiple Bernoulli model rather than a multinomial model that is more commonly used in IR.

- (a) How does this model take term absence (in the query) into account and what difference might it make? Illustrate with an example.
  - (b) In what way does this model fail to take repetitions (in the query) into account and what difference might that make? Illustrate with an example.
4. A major disadvantage of Equation 2 is that it is a product over all terms in the vocabulary, not just over terms in the query. At first glance, this seems to mean that inverted files will not provide any efficiency. Describe some pre-calculating and manipulation that you could do to implement this efficiently. *Hint:  $A = A * B / B$  or, if you use logs,  $A = A - B + B$ .*
  5. Why do most language modeling implementations calculate, store, and manipulate the logarithms of probabilities?
  6. What does it mean to say that a language model is *generative*? Is the vector space model generative? Why or why not?

## Problem EV (evaluation)

1. In the class project, several queries were generated and run to find top-ranked documents for each topic. A pool for judging a particular topic was developed by selecting the documents at rank 1 for every submitting system, then those at rank 2, 3, and so on, until there were at least 120 unique documents found (recall that the sets of top-ranking documents probably overlap).

Those 120 documents were then sorted by document identifier, divided into six sets (top 20, next 20, ..., and bottom 20 in alphabetical order), and handed out to six people to be judged (20 documents each). Suppose that an individual fails to return his or her 20 documents, forcing us to assume that they are all non-relevant.

What will be the impact on the evaluation measures for that topic? How significant is the impact? Is any system likely to be more impacted than another? Focus on recall, precision, and average precision, though you may discuss other measures if you like.

2. Even if all 120 documents are judged, there are still hundreds of documents that were not judged for that topic. What impact does *that* have on evaluation scores? Again, focus on recall, precision, and average precision.
3. Mean Reciprocal Rank (MRR) is an evaluation measure used for evaluating systems when there is precisely one relevant document. Each topic is given the score  $1/r$  where  $r$  is the rank at which the relevant document was found. In this situation, what are the pros and cons of MRR as opposed to mean average precision?
4. The lecture notes present one method for interpolating precision values for recall values between measured recall points. The approach is to take the maximum precision value “to the right” of the recall point in question. Specifically:

$$P(R) = \max\{P' \mid R' \geq R \wedge (R', P') \in S\}$$

It would also be possible to interpolate by drawing straight lines between the measured points. This was called “connect the dots” in the “revised” evaluation lecture notes. Argue either against or for “connect the dots” in contrast to the “max to the right” interpolation described above.

### Problem RF (relevance feedback)

1. Why might “pseudo-relevance feedback” sometimes transform an imperfect query into one that is incredibly bad?
2. Suppose a query had precisely one relevant document in a collection. Speculate on whether or not “local feedback” (of any type) would be likely to help. Support your speculation.
3. In a sense, LSI builds a global association thesaurus that implicitly causes query expansion (except that it’s done at indexing time). In what ways is the “expansion” caused by LSI different from that caused by “local feedback” at query time? What impact is that likely to have on effectiveness?
4. Consider the following equations that describe how the Local Context Analysis approach weights candidate expansion concepts with respect to a query (the same equations appear in the lecture notes):

$$f(c, Q) = \prod_{w_i \in Q} (0.01 + \text{co\_degree}(c, w_i))^{\text{idf}(w_i)}$$

$$\text{co\_degree}(c, w) = \max\left(\frac{n_{cw} - \text{En}(c, w) - 1}{n_c}, 0\right)$$

$$\text{En}(c, w) = \frac{n_w n_c}{N}$$

$$\text{idf}(w) = \min(1.0, \log_{10}(N/n_w)/5)$$

What is the purpose of the  $\min()$  operator on the  $\text{idf}$  function? (The lecture notes indicate it is to “cap the IDF component”, but that’s an error.) In answering this question, you should indicate how often a term must occur in order to have its IDF component be higher than one, and why it might not make sense for it to become smaller than one.