CS6740/INFO 6300: Advanced Language Technologies, Spring 2010
1/28/10: Example problem for lecture-guide preparation

Recall from the "course description and policies" handout that lecture-guide preparation is part of the coursework for this class, and that lecture guides consist of "roughly textbook-quality 'scribe notes"' plus one or more original worked problems. See the "course description and policies" handout for precise details, and websites for previous runnings of this course for samples (e.g., `http://www.cs.cornell.edu/courses/cs674/2007fa`).

To give you some guidance on what expectations might be for your worked problems, below is an example of a question (solution omitted) that is a "finger exercise" in terms of the difficulty of the problem, but exemplifies the notion of a "deeper" question because it involves an exploration of the underlying motivations behind a concept introduced in class — it asks the reader to consider the question, "what if we tried a seemingly reasonable alternative instead?" One goal of this class is to foster this kind of thinking, since asking such questions can often lead to very interesting research results.

Also note that this question is longer than it needs to be, and includes more guiding of the reader than is necessary for a graduate class, but it would be a fine submission (with solution, of course) — (a) would count as the "finger exercise", and (b) and (c) and the surrounding exposition cover the "deeper question" requirement.

Sample question:

Recall that the motivation behind incorporating inverse document frequency into term weights was that terms occurring in many documents shouldn't be as important when it comes to distinguishing one document in the collection from another. However, it's natural to ask why we should necessarily use an *inverse* (reciprocal) functional form. Hence, in this problem we consider a potential alternative.

For the purposes of this question (omitting logs makes hand calculation easier), define $\mathrm{IDF} = n/\mathrm{docfreq}_i$, and define the *negative document frequency (*NDF*)* of the i^{th} term as

$$\mathrm{NDF}_i = n - \mathrm{docfreq}_i + 1,$$

where n is the number of documents in the corpus and $\mathrm{docfreq}_i$ is the number of documents that contain the ith term. Note that the IDF and the NDF are both equal to 1 for terms that occur in all n documents, and that both quantities are equal to n for terms that occur in exactly one document.

Now suppose we have a 100-document corpus with an index containing only five terms, where these terms have the following $\mathrm{docfreq}$ values:

i	1	2	3	4	5
w_i	*aerospace*	*civil*	*computer*	*engineering*	*science*
$\mathrm{docfreq}_i$	50	50	2	20	20

Some quick self-checks: using the simplest tf-idf weighting method with cosine normalization, the document "computer science computer engineering" would be converted to the vector

$$(0/\sqrt{10050}, 0/\sqrt{10050}, 100/\sqrt{10050}, 5/\sqrt{10050}, 5/\sqrt{10050}).$$

Also note that summing the $\mathrm{docfreq}$ values does not give the size of the corpus (100); can you see why this might be the case?

Anyway, let's continue. Let the first two documents d_1 and d_2 be the following:

(OVER)

d_1:	engineering civil engineering aerospace engineering civil engineering
d_2:	computer science

Finally, suppose the user has supplied the query $q =$ "computer engineering", which we translate to the unnormalized vector $\overrightarrow{q} = (0, 0, 1, 1, 0)$. We assert that given the document-frequency information above, humans would prefer document d_2 over d_1 with respect to this query, even though both documents contain exactly one term in common with q.

(a) Compute the document vectors $\overrightarrow{d}_1^{(\mathrm{IDF})}$ and $\overrightarrow{d}_2^{(\mathrm{IDF})}$ that result from applying the tf-idf method to d_1 and d_2, showing your work. Which document (or is it a tie) is ranked more relevant to the query (using the inner product as scoring function)? Explain.

(b) Compute the document vectors $\overrightarrow{d}_1^{(\mathrm{NDF})}$ and $\overrightarrow{d}_2^{(\mathrm{NDF})}$ that result from applying the tf-idf method to d_1 and d_2, *except* replacing all appearances of IDF quantities with NDF quantities, showing your work. Which document (or is it a tie) is ranked more relevant to the query? Explain.

(c) We claim that you should have gotten different rankings for the above two subproblems. Explain why the difference occurs, and whether using NDF instead of IDF was beneficial or harmful.

Note: We are fully aware that the NDF is a linear function and the IDF is hyperbolic. Your answer should specify *why* these different functional forms impact the retrieval results.