SIMS 202 Assignment 7
IR Weighting and Ranking
Due Tuesday November 18

Assignment authors, Marti Hearst and Ray Larson

Please bring a hardcopy of your assignment to class on Tuesday, Nov. 18. There are three questions in total, with sub-parts to each one.

1 Practice with Sigma Notation

Recall the meaning of sigma (summation) notation. For example,

$$n = 10; \qquad s = \sum_{i=0}^{n-1} i$$

means s gets assigned the sum of all the integers from 0 to 9, inclusive, or $0 + 1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 = 45$. The index is i and its boundaries are from 0 to $n - 1$. As another example

$$n = 3; \qquad s = \sum_{i=1}^{n} a_i * a_{i+1}$$

means s is assigned the sum of $a_1 * a_2 + a_2 * a_3 + a_3 * a_4$. And

$$n = 3; \qquad s = \sum_{i,j=1}^{n} a_i * b_j$$

means s is assigned the sum of $a_1 * b_1 + a_2 * b_2 + a_3 * b_3$.

For the problems below you should use a calculator or computer. You may want to show the main intermediate stages of the computation if you're unsure about how to do the work.

Compute s for the following three formulas (be sure to check the boundaries for the indices).

(a) $n = 12; \qquad s = \sum_{i=0}^{n-1} i^2$

(b) $m = 8; \qquad s = \sum_{j=1}^{m} -j$

(c) $n = 4; \qquad a_i = i; \qquad b_j = j + 3; \qquad s = \sum_{i,j=0}^{n-1} a_i * b_j$

2 Computing Term Weights

For a collection C consisting of N documents, consider the following term weight formulae:

$w_{ik} = tf_{ik} * idf_k$
$idf_k = log(N/n_k)$

where

T_k = term k in collection C
tf_{ik} = frequency of term T_k in document D_i
n_k = the number of documents in C that contain term T_k
f_k = the total frequency of term T_k in all documents of C
M = the number of unique terms in C
idf_k = inverse document frequency of term T_k in collection C
w_{ik} = the weight of term T_k in document D_i

(If you write a program to calculate these or the following questions, please include the source code with the answers you hand in.)

(a) It is always the case that $f_k \geq n_k$? Briefly explain why.

(b) Say the term "user" occurs in the document D_1 twelve times and in five documents in the entire collection C, and that the collection consists of 100 documents. What is the weight of this term in this document?

Now say for the same collection the term "user" occurs in the document D_2 one time. What is the weight of this term in this document?

Be sure to show your work.

3 Computing Document Similarity

(a) Assume the documents D_1, D_2, and D_3, which have the following characterstics:

Document D_1 contains "information" 20 times and "retrieval" 3 times.

Document D_2 contains "information" 1 time and "retrieval" 15 times.

Document D_3 contains "information" 12 times and "retrieval" 10 times.

Also assume that

“information” occurs in 120 documents in the collection

“retrieval” occurs in 70 documents in the collection

N is 1000

Draw a graph showing the vectors for the raw frequency counts. Place “information” on the x-axis and “retrieval” on the y axis. (You may use any graphing tool that you wish, hand-drawn graphs SHOULD use rulers).

(b) Assume the query consists of the two words “information” and “retrieval”. Compute the similarity value between the query and each of the documents D_1, D_2, and D_3.

To compare the similarity of two documents, or a document and a query (since the query is treated the same way as a document) use the weighting formula below to compute each w_{ik} and the following similarity comparison formula.

(This weighting formula normalizes the term weights.)

$$w_{ik} = \frac{tf_{ik} * log(N/n_k)}{\sqrt{\sum_{k=0}^{M-1} (tf_{ik})^2 * [log(N/n_k)]^2}}$$

$$sim(D_i, D_j) = \sum_{k=0}^{M-1} w_{ik} * w_{jk}$$

(Hint: if a term does *not* occur in a document or query then its weight is zero, so it can be ignored in the calculation. Also note that in IR log is USUALLY base 2 logarithms, but in this case you can use the natural logarithm or base 10 logarithm if that is all that is available on your calculator – please indicate which you are using)

Be sure to show your work. Discuss the results briefly.

(c) Draw a graph showing the normalized vectors for the documents (represent the documents in terms of their normalized weights). Place “information” on the x-axis and “retrieval” on the y axis. Also draw the vector for the query. Does the graph correspond with your results for part (b)? How is this related to part (a)?

(d) What would the results above look like if we just used tf for the term weights, without multiplying by idf? What would they look like if “information” had occurred in 400 documents in the collection instead of 120?