INFO 202 – ASSIGNMENT 8
Assigned 11-19-08; due 12-01-08

Term Weighting and Ranking Calculations

PRACTICE - SIGMA NOTATION

Recall the meaning of sigma notation. For example:

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

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; $s = \sum_{i=1}^{n} a_i \cdot a_{i+1}$

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

n = 3; $s = \sum_{i,j=0}^{n} a_i \cdot b_j$

means *s* is assigned the sum of $a_0 * b_0 + a_1 * b_1 + a_2 * b_2 + a_3 * b_3$.

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

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

(a) *n* = 6; $s = \sum_{i=1}^{n-1} 3\mathrm{i}$

(b) $m = 5$; $$s = \sum_{j=1}^{m} 2^{j}$$

(c) $n = 7$; $a_i = i + 1$; $b_j = 2j$; $$s = \sum_{i,j=1}^{n-1} a_i^2 + b_j^2$$

2. COMPUTING TERM WEIGHTS

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

$$w_{ik} = tf_{ik} \cdot idf_k$$

$$idf_k = \log\left(\frac{N}{n_k}\right)$$

where

T_k = term *k* in collection C

tf_{ik} = frequency of term T_k in document D_i

Note this term represents one value, not two as the sub i and sub k seems to indicate.

n_k = number of documents in C that contain term T_k

f_k = total frequency of term T_k in all documents of C

M = total number of unique terms in C
N = total number of documents in C
idf_k = inverse document frequency of term T_k in collection C
w_{ik} = the weight of term T_k is document D_{ik}

2a. Why do we need two different variables -- tf and idf -- in the calculation of term weights?

2b. What is the relationship between the values of M and N?

2c. For a given collection, different search engines might use different values of M. Why?

COMPUTING DOCUMENT SIMILARITY

Be sure to show your work.

Assume the documents D_1, D_2, and D_3, have the following characteristics:

- Document D_1 contains "user" 12 times and "interface" 3 times.
- Document D_2 contains "user" 5 times and "interface" 16 times.
- Document D_3 contains "user" 8 times and "interface" 7 times.
- "User" and "interface" are the only words that D_1, D_2 and D_3 contain.

Remember, if a term doesn't occur in a document or query then its weight is zero. In this case, we are comparing the query terms to the document, so at most there are 2 terms to consider.

Also assume that:

- "user" occurs in 120 documents in the collection
- "interface" occurs in 60 documents in the collection
- The number of documents in the collection, *N,* is 5000.

3a. Draw a graph showing the vectors for the raw frequency counts. Place "user" on the x-axis and "interface" on the y axis.

3b. Assume the query consists of the two words "user" and "interface". 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 (where the query is viewed 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.)

$$w_{ik} = \frac{tf_{ik} \cdot idf_k}{\sqrt{\sum_{n=0}^{M-1} \left((tf_{ik})^2 \cdot (idf_k)^2 \right)}}$$

$$sim(D_i, D_j) = \sum_{n=0}^{M-1} \left(w_{ik} \cdot w_{jk} \right)$$

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

4. Vector Graphs

a. Draw a graph showing the normalized vectors for the documents (represent the documents in terms of their normalized weights from your work in part (3b)). Place "user" on the x-axis and "interface" on the y axis. Also draw the vector for the query.

b. Does the graph correspond with your results for part (3b)? How is this related to part (3a)?

c. What would the results above look like if we just used tf for the term weights, without multiplying by idf?

d. What would the results above look like if "user" had occurred in 600 documents in the collection instead of 120?