CS/ENGRI 172, Fall 2003: Computation, Information, and Intelligence
10/22/03: Vector Space Models

Quantities Related to Terms and Documents

Given a word w_i from the vocabulary V of w_1 to w_m and a corpus D of documents $d_1, \ldots, d_n$ we use the following measures when ranking documents. (Generally, we will use the variable i to index terms and the variable j to index documents.)

- *term-document frequency*: $\text{freq}_{i,j}$ the number of times term w_i occurs in document d_j
- *document frequency*: docfreq_i the number of documents that contain term w_i
- *inverse document frequency*: IDF_i $n/\text{docfreq}_i$

Self Check: To test that you understand how to compute these quantities, consider d_1 and d_2 given on your handout from 10/8/03 ("Information and Intelligence"). If we consider the words w_1 = "he", w_2 = "the" and w_3 = "we", assuming these are the only two documents in our corpus, you should get the following values for this set of quantities:

$$\begin{array}{llll} \text{freq}_{1,1} = 1 & \text{freq}_{1,2} = 1 & \text{docfreq}_1 = 2 & \text{IDF}_1 = 1 \\ \text{freq}_{2,1} = 3 & \text{freq}_{2,2} = 1 & \text{docfreq}_2 = 2 & \text{IDF}_2 = 1 \\ \text{freq}_{3,1} = 2 & \text{freq}_{3,2} = 0 & \text{docfreq}_3 = 1 & \text{IDF}_3 = 2 \end{array}$$

Vector Length Normalization

We use vectors to represent our documents and queries. It is a handy fact that we can *normalize* any vector of non-zero length to get a new vector that points in the same direction, but has unit length. Recall that for any vector $\vec{x} = (x_1, x_2, \ldots, x_n)$, the vector length of $\vec{x}$ is $\sqrt{\vec{x} \cdot \vec{x}} = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$. So we can normalize a vector by simply dividing each of its components by the vector's length L as a whole. The resulting vector will then have length 1:

$$\begin{aligned} \text{length}((x_1/L, x_2/L, \ldots, x_n/L)) &= \sqrt{x_1^2/L^2 + x_2^2/L^2 + \cdots + x_n^2/L^2} \\ &= \frac{1}{L}\sqrt{x_1^2 + x_2^2 + \cdots + x_n^2} \\ &= \frac{1}{L}\text{length}(\vec{x}) = \frac{1}{L}(L) = 1 \end{aligned}$$

Furthermore, we are guaranteed that the normalized vector will have the same directionality as the unnormalized vector:

$$\begin{aligned} (x_1, x_2, \ldots, x_n) \cdot (x_1/L, x_2/L, \ldots, x_n/L) &= x_1^2/L + x_2^2/L + \ldots + x_n^2/L \\ &= \frac{x_1^2 + x_2^2 + \ldots + x_n^2}{L} \\ &= \frac{x_1^2 + x_2^2 + \ldots + x_n^2}{\sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}} \\ &= L = \text{length}(\vec{x}) \end{aligned}$$

so the cosine of the angle between the vectors must equal 1.

IR with Vector Space Models
Various types of vector space models for IR vary based on how the document vector $\vec{d_j}$ is constructed from a document d_j. For all of these schemes, our IR system will build a document vector $\vec{d_j}$ for each document d_j, build an unweighted, unnormalized query vector $\vec{q}$ from the query q (this will just be ($\text{freq}_{1,q}$, $\text{freq}_{2,q}$, $\ldots \text{freq}_{m,q}$), the frequency of each term in the query). Ranking is then performed by taking the dot product of a query vector with each of the document vectors.

Term-frequency Weighting
In this scheme, we set the document vector $\vec{d_j}$ for document d_j as follows:

$$\vec{d_j} = (\text{freq}_{1,j}/N_j, \text{freq}_{2,j}/N_j, \ldots, \text{freq}_{m,j}/N_j)$$

where $N_j = \sqrt{\sum_{i=1}^{m}(\text{freq}_{i,j}^2)}$ is the length-normalization factor.

Tf-idf Weighting
For better ranking, we would like to take into account not just statistics across a single document, but across the corpus of documents as a whole. To reflect the overall content of our corpus, we can combine term-frequency and inverse-document-frequency and set a document vector $\vec{d_j}$ for document d_j to be:

$$\vec{d_j} = (\text{freq}_{1,j}\text{IDF}_1/N_j, \text{freq}_{2,j}\text{IDF}_2/N_j, \ldots, \text{freq}_{m,j}\text{IDF}_m/N_j)$$

where $N_j = \sqrt{\sum_{i=1}^{m}(\text{freq}_{i,j}\text{IDF}_i)^2}$.