Chapter No. 6 - Scoring, term weighting and the vector space model (Article 6.2-6.4)
Chapter No. 7 – Computing scores in complete search System (Only Article 7.1)

<Food for Thoughts>

1. What do we mean by Vector Space Model? Explain with an example.
2. Consider a corpus C that consists of the following three documents:
   D1: dil dil Pakistan, jan jan Pakistan
   D2: Pakistan hum sub ki jan
   D3: dil aur jan Pakistan Pakistan
   Assuming that the term frequencies are normalized by the maximum frequency in a given document, calculate the TF-IDF weighted term vectors for all documents in C. Assume that the words in the vectors are ordered alphabetically?
   For the above corpus C, consider a query "dil jan Pakistan". Calculate the TF-IDF weighted query vector for this query.
   Using the cosine similarity measure, calculate the similarity of the query q with all documents in the collection. Assume that term frequencies are normalized by the maximum frequency in given query.
3. Consider the given weights g1 = 0.2, g2 = 0.31 and g3 = 0.49, what are all the distinct score values a document may get?
4. When can IDF value of a term be zero? Explain?
5. Why is the idf of a term always finite? What is the idf of a term that occurs in every document? Compare this with the use of stop word lists.
6. Why tf*idf weighting is giving best results in VSM? Explain.
7. Why normalization is required for term frequency? What are some different normalizations schemes for term frequency?
8. What do we mean by Champion List? How it facilitates quick relevant scoring of documents?
9. What do we mean by Static Quality Score for a document?
10. When discussing champion lists, we simply used the r documents with the largest tf values to create the champion list for t. But when considering global champion lists, we used idf as well, identifying documents with the largest values of g(d) + tf-idft for a document d. Why do we differentiate between these two cases?
11. Explain how the common global ordering by g(d) values in all high and low lists helps make the score computation efficient.