The implementation is based on the concept of BM25.

The BM25 implemented in my program is based on information obtained from the implementation.pdf file provided.

The formula used.

$$\sum_{i \in Q} \log \frac{(r_i+0.5)/(R-r_i+0.5)}{(n_i-r_i+0.5)/(N-n_i-R+r_i+0.5)} \cdot \frac{(k_1+1)f_i}{K+f_i} \cdot \frac{(k_2+1)qf_i}{k_2+qf_i}$$

- k_1, k_2 and K are parameters whose values are set empirically
- $K = k_1((1-b) + b \cdot \frac{dl}{avdl})$ *dl* is doc length
- Typical TREC value for k_1 is 1.2, k_2 varies from 0 to 1000, b = 0.75

! ri is the # of relevant documents containing term i

! (set to 0 if no relevance info is known)

! ni is the # of docs containing term i

! N is the total # of docs in the collection

! R is the number of relevant documents for this query

! (set to 0 if no relevance info is known)

! fi is the frequency of term i in the doc under consideration

! qfi is the frequency of term i in the query

! k1 determines how the tf component of the term weight changes as fi increases.

(if 0, then tf component is ignored.)

Typical value for TREC is 1.2;

so fi is very non-linear (similar to the use of log f in term wts of the vector space model)

--- after 3 or 4 occurrences of a term, additional occurrences will have little impact.

! k2 has a similar role for the query term weights.

Typical values (see slide) make the equation less sensitive to k2 than k1 because query term frequencies are much lower and less variable than doc term frequencies.

! K is more complicated. Its role is basically to normalize the tf component by document length.

! b regulates the impact of length normalization. (0 means none; 1 is full normalization.