An Experimental Study of Factors Important in Document Ranking

Donna Harman

Lister Hill National Center for Biomedical Communications
National Library of Medicine
Bethesda, Maryland, 20209

Abstract

The ability to effectively rank retrieved documents in order of their probable relevance to a query is a critical factor in statistically-based keyword retrieval systems. This paper summarizes a set of experiments with different methods of term weighting for documents, using measures of term importance within an entire document collection, term importance within a given document, and document length. It is shown that significant improvements over no term weighting can be made using a combination of weighting measures and normalizing for document length.

1. Introduction

There is considerable interest in retrieving information from existing sources as different as semi-structured databases and unstructured document collections, without having to re-organize the data and without requiring a complicated syntax for submitting queries. A statistically-based keyword system, using automatic indexing of the database and natural language queries, is a very attractive choice, offering ease of implementation, minimum modifications to the database, and availability to users of natural language. A major element in the success of these keyword systems is their ability to rank the retrieved items in order of probable relevance to the query. Consequently, the choice of the factors and the weightings of these factors in the ranking algorithm are critical to retrieval effectiveness. The experiments described in this paper were aimed at identifying the factors important in ranking and at examining how to most effectively combine them.

These experiments were done using the IRX testbed put into operation in 1985 at the Lister Hill National Center for Biomedical Communications, the research arm of the National Library of Medicine.

The IRX (Information Retrieval Experiment) project was started in 1984 to answer two major questions. First, given the wealth of experimentation done using statistically-based keyword retrieval, what are the upper bounds of performance, and what new techniques can be developed to extend these bounds to those projected for expert systems. Included in this question is an estimate of the cost required to attain this higher level of performance in terms of software development, maintenance, and user effort. Second, what are the basic parameters in a traditional statistically-based keyword retrieval system, and what kinds of changes in these parameters are needed to match different document collections or parsing techniques. For example, is it possible to directly tune a retrieval system to perform effectively given a set of database characteristics or a particular parsing method?

Research is currently being done in two major areas. First, collaboration with the Welch Medical Library at the Johns Hopkins School of Medicine will result in both a new test collection and the opportunity to allow users to access a collection of medical text online. At a later stage of research there are plans to have users test intelligent interfaces that interact with the user to refine search requests. The second area involves the establishment of a baseline level of performance, and the investigation of the factors important in a statistically-based keyword retrieval system. In this area, ranking techniques have been selected as the first field of investigation, and the results of this research form the basis for this paper.

2. Current ranking techniques

Five different components in the ranking of documents have been reviewed by McGill: 1) the form of document representation, 2) the weighting of the document terms, 3) the form of query representation, 4) the weighting of the query terms, and 5) the similarity measure [MCGILL79]. Of these five components, only the weighting of the document terms was investigated in this experiment; the others were held constant. Automatic indexing using full words was used as the form of document (and query) representation, binary weighting of the query terms was used, and the similarity measure used was an inner product.

Examining the weighting of document terms raises two major questions: 1) what factors in a document are important in measuring document-query similarity, and 2) how should these factors be measured and combined. In reviewing past research, four major factors emerged: 1) the number of term matches between the query and a document, 2) the importance of a given term within a document collection, 3) the importance of a given term within a given document, and 4) the length of a document.

2.1 The number of term matches between a query and a document

The number of term matches between a query and a document is an elementary measure of document-query similarity, in which the rank of a document is based on the number of document and query terms that match. This measure, which can be viewed as a binary weighting of the document terms, has been used as a baseline for comparing different ranking methods [CROFT82]. The technique works better than no ranking, but fails to discriminate between terms of varying importance, or between documents that provide more matches just because they are

2.2 The importance of a term within the entire document collection

Many functions have been developed for measuring the importance of a term within an entire document collection. One of the first, the inverse document frequency [SPARCK JONES72;SALTON83], is based on the number of documents in which a given term k occurs in a document collection. Specifically;

$$inverse\ document frequency_k = \log_2 \frac{N}{NumDoc_k} + 1$$

where N = number of documents in document collection
$NumDoc_k$ = number of documents in the collection that contain one or more instances of term k

A second measure, noise, [DENNIS64;SALTON83] also measures term occurrence within a collection, but measures the concentration of that term rather than occurrence counts. Specifically

$$noise_k = \sum_{i=1}^{N} \frac{Freq_{ik}}{TotFreq_k} \log_2 \frac{TotFreq_k}{Freq_{ik}}$$

where N = number of documents in document collection
$Freq_{ik}$ = the frequency of term k in document i
$TotFreq_k$ = the total frequency of term k in the document collection

If a term appears only in one document, the noise is zero, but for terms that are fairly evenly distributed within a document collection, the noise is much higher.

Both inverse document frequency weight and noise were investigated in the present set of experiments. Other measures of term importance within a collection were not investigated: 1) the term discrimination measure [SALTON73], 2) the term precision value [YU82], and 3) the 2-Poisson model [HARTER75;RAGHAVAN83]. These measures involved a level of computational complexity that was beyond the scope of the current investigation.

2.3 The importance of a term within a given document

The importance of a term within a given document is usually expressed as a function of its frequency within that document. Two functions were investigated in this study: 1) the raw frequency, and 2) the $\log_2$ of frequency.

2.4 Normalization for document length

The factor of document length is not generally directly used, but becomes involved in the way the factors are combined (such as the use of a cosine similarity measure). This factor was used directly as a final normalizing function in the latter set of experiments.

2.5 Combining the measures

The various ranking factors can be combined in many ways to create a total term weight for a given document. Often, the two types of term importance factors--importance within a collection and importance within a document--are multiplied to create a single term weight [SALTON83]. This weight is used in place of a binary weight (present or not present) in one of many similarity measures [MCGILL79]. Alternatively, the various factors can be combined additively, with constants used to weight the importance of each factor [CROFT79]. Both ways of combining factors were investigated in these experiments.

3. Methodology

3.1 Experimental Methods

The test collection used was the Cranfield collection with 225 queries and 1400 documents. The documents and queries were indexed using full words and a standard common word list. The terms in the query were treated as having binary weights, and the weighting factors applied only to the document terms. The inner product of query terms and document terms was used as the similarity measure.

Using batch mode, each query was parsed into noncommon terms, and a list of all documents containing one or more of these terms was input to the ranking routine. Note that since the indexing method is constant, the same list of documents was retrieved for all ranking methods, and therefore the experiments compared only the ranking of a given set of retrieved documents.

The experiments were run in the following order:

a) investigation of the use of single measures alone in the weighting of document terms: 1) the number of term matches, 2) the within-document frequency, 3) the $\log_2$ of the within-document frequency, 4) the inverted document frequency weight measure, and 5) the noise measure;

b) investigation of additively combining the number of matches with one of the other four measures to form a term weight;

c) investigation of additively combining one of the measures of the importance of a term within a given document (within-document frequency and $\log_2$ within-document frequency) with one of the measures of the importance of a term within a document collection (inverted document frequency weight and noise) to form a term weight;

d) investigation of multiplicatively instead of additively combining the measures described in c) to form a term weight;

e) investigation of normalizing the resulting similarity value for a given document by the length of the document.

3.2 Evaluation

Three methods of evaluation are presented in the tables. First, the standard recall and precision measures are given, with the averaging based on that done by the SMART system. The percentage of improvement in average precision at three given recall levels is used to compare methods. Additionally the average rank recalls are given and the E measures are calculated for 10 and 30 document cutoffs. Further description of these evaluation methods can be found in [SALTON83]. All relevant documents not retrieved for a given query are assigned ranks of 1400, 1399, etc. The Cranfield documents that are considered "source" documents for a given query are not counted as retrieved (either relevant retrieved or nonrelevant retrieved) for that query.

4. Results

4.1 Single document-query similarity measures (Table 1)

Five single measures were run: 1) the number of matches, 2) the raw frequency of a term within a document, 3) $\log_2$ of the frequency of a term within a document, 4) the inverted document frequency weight measure, and 5) the noise measure.

The first, the number of matches (Run 1, Table 1), was used as the baseline for comparison of the other methods. Examination of the ranking of the documents illustrates the problems of this method of ranking. For example, query 12, with five relevant documents, ranks these at ranks 7, 72, 91, 480, and 483. One of the relevant documents has 4 terms that matched the query, two others had 3 matching terms, one had 2 matching terms, and the last relevant had only 1 matching term. The terms involved have a range of number of postings (the number of documents within the collection containing these terms) from 703 for the term "flow" to 15 for the term "machines", yet all terms are treated equally--documents containing only "flow" are randomly ordered within all documents containing only "machines". Moreover, documents containing multiple occurrences of a term are ranked no higher than ones containing only a single instance of the term.

The second and third measures, within-document frequency and $\log_2$ of within-document frequency, are based on the frequency of the matching terms within a document. The weighting for a document term is the frequency (or $\log_2$ of the frequency) of the term within the document being ranked. Again, with respect to query 12, the five relevant documents rank at 40, 103, 223, 351, and 965 using the raw within-document frequency of a matching term as its weight. Using the $\log_2$ of that frequency, the ranks are 28, 71, 76, 229, and 900. Comparing this performance for query 12 to that for using only the number of matches shows a decline in performance using the raw frequency, and a slight improvement for $\log_2$ of the frequency. This type of performance is reflected in the averages over 225 queries (Runs 2 and 3, Table 1). Using the raw frequency alone, the performance is 22% worse than the baseline simple matching. When the $\log_2$ of the frequency is used instead of the simple raw frequency, the performance improves slightly, showing an average improvement in precision of 6% over simple matching. The other evaluation measures verify these findings. The explanation for the performance difference lies in that fact that using the raw frequency measure alone allows terms of high frequency in a document (3 or more occurrences) to dominate the similarity measure. A term appearing 3 or 4 times should usually not have the same importance as 3 or 4 matching terms appearing once. By using the $\log_2$ of the frequency, this effect is lessened, and the within-document frequency becomes a more reasonable measure.

The fourth measure, the inverted document frequency weight, (measured by the inverse of the number of postings, see equation 1), takes the importance of a term within a document collection into account. The weighting for a document term is simply the inverted document frequency weight of the term. This weighting method improves the ranks of the relevant documents for query 12 to 4, 5, 30, 125, and 515. For example, relevant document 649 moves from rank 91 to rank 5, because two of the three matching terms have high inverted document frequency weights, based on their relatively low number of postings (31 for "ground" and 5 for "machines"). This type of improvement, although not uniformly better for all queries or documents, averages to a 20% improvement in average precision over the method using only the number of matches (see Run 4, Table 1). All other evaluation measures also show significant improvement.

The fifth measure, noise, also measures the importance of a term within a document collection, but is based on the distribution of the term throughout the collection and within each document (see equation 2). The weighting of a document term is a normalization of the noise of that term. The noise measure needs to be normalized since the importance of a term is in inverse relation to its noise. Normalization was done by subtracting the actual noise of a term from the maximum possible noise in the collection (9.43 for the full-word-indexed Cranfield 1400 collection), so the normalized noise ranges from 0.00 (very noisy term) to 9.43 (very low noise). Examining the ranking of the relevant documents in query 12, there is still more improvement, up to ranks 3, 4, 28, 73, and 383. When the words in the retrieved documents are examined, it appears that the noise measure provides a fine tuning on the inverted document frequency weight measure by taking into account the distribution within documents, not just within the collection. For example, relevant document 194 moves from rank 515 to 363, based on the single matching word "calculated". This term has a medium number of postings (153), leading to a medium value of inverted document frequency weight. Since, however, the additional occurrences of the word tend to be concentrated in few documents, the normalized noise is relatively higher than the inverted document frequency weight, and this gives the improvement in rank. A second example in the same query is the word "flow", which has a high number of postings (703), but an even higher frequency (1854) with a very even distribution, leading to an extremely low normalized noise. This causes many non-relevant documents that had previously been ranked higher, based on the inverted document frequency weight measure, to be lowered in probable importance using the normalized noise measure. It should be noted that because of individual word idiosyncrasies, the normalized noise measure does not always improve performance on a query-by-query basis, but offers a slight improvement (4%) in average precision over the inverted document frequency measure, and a very significant improvement (24%) over the baseline method (see Run 5, Table 1). This is verified by the other evaluation measures, with the improvement being greatest at low levels of recall.

4.2 Combination I: Combining the number of matches with other single measures (Table 2)

Combining the number of matches with each of the other single measures produced interesting results. The weighting of a document term is a constant C_1 plus a second constant C_2 times the term weight generated by the measure being examined. Combining the number of matches with the two measures of frequency-within-document gives significant improvement (Runs 2 and 3, Table 2). When the number of matches was combined with the two measures of term importance within a collection (inverted document frequency weight and normalized noise), there is no significant change in performance (Runs 4 and 5, Table 2). The combining of the number of matches and the inverted document frequency weight was used by Croft [CROFT79] in working with a manually-indexed version of the Cranfield 1400, with slightly better results. As expected, if the constants C_1 or C_2 are increased from 1, performance approaches that of the measure being more heavily weighted (results not shown). These findings are verified by the other evaluation measures. The performance difference is due to the nature of the two types of measures--the two within-document frequency measures vary from 1 to the maximum (or $\log_2$ of the maximum) frequency for each term, applying additional weight for higher frequency terms. Adding the number of matches to this type of term-importance measure gives more weight to the number of matches, not just the frequency of the terms. Adding the number of matches to either the normalized noise or the inverted document frequency weight just increases each term weight by the same amount, producing little relative difference in ranking. These results suggest the reasonable combination of a frequency-within-document measure and a measure of term importance within an entire collection.

4.3 Combination II: Additively combining the single measures for term importance within a collection and term importance within a given document (Table 3)

The single measures were additively combined in a way similar to combination I. The weighting of the document terms is a constant C_1 times the first measure plus a second constant C_2 times the second measure. Only one measure for term importance within a given document was used ($\log_2$ of the within-document frequency), since using the raw frequency was shown to be a significantly worse measure. Both measures for term importance within a document collection were tried. Several types of results can be seen in Table 3. First, the effects of additively combining the two types of term importance measures is roughly additive (6.0 + 19.9 for inverted document frequency weight and 6.0 + 23.9 for normalized noise), with the normalized noise measure showing the greatest gain (Runs 5 and 6, Table 3). This is an indication of the complementary nature of the two types of measures--the effects do not mask each other. The second finding apparent from Table 3 is the continued slight superiority of the normalized noise measure--adding the within-document frequency has widened the performance difference. Third, the weighting of the two types of measures seems to be best at equal weighting, but other weightings make only slight decrements in performance (Runs 7 and 8, Table 3). These results also hold true for the inverted document frequency weight measure although the data is not included in Table 3. Adding the number of matches causes a slight, but insignificant decline in performance (data not included in Table 3). These results are verified by the other evaluation measures.

4.4 Combination III: Multiplicatively combining the single measures for term importance within a collection and term importance within a given document (Table 3)

The single measures were multiplicatively combined in a manner similar to combination II. The weighting of the document terms is the product of the first measure and the second measure. Table 3, Runs 9 and 10, shows this method of combining single measures, leading to a 29% and 35% improvement over simple matching, to be better than additive combinations. The results are consistent with those for additive combinations--adding the number of matches makes no significant improvement (data not included in the table), and the normalized noise measure still performs more effectively than the inverted document frequency measure. All these results are verified by the other evaluation measures.

4.5 Modification for length (Table 4)

Documents that are significantly longer than the average document length in the collection can have a higher rank simply because they are longer. This affects the rank both by increasing the total number of matches, and by causing higher frequencies of the matching terms. The effects of document length are usually handled indirectly by the use of a cosine similarity measure. As the necessary document statistics were not readily available in the current IRX system, this was not possible. Additionally, directly using the factor of document length allowed a clearer insight into the effects of document length. Two measures of document length were used, analogous to the two measures of within-document frequency--the actual length of a document (as measured by the total number of characters needed to store the document) and the $\log_2$ of the actual length. The length modification used was a division of the term weight of a document by its length (or the $\log_2$ of its length). Table 4 shows the effects of using document length to modify the document weights for the additive and multiplicitive combinations of measures. As can be seen, modification by the raw document length does not improve performance over the combinations (Run 3, Table 4--other combinations not shown), but modification by the $\log_2$ of the document length improves performance very significantly, leading to a maximum improvement of 44% over simple match (Runs 4, 5, 6, and 7, Table 4). Using the $\log_2$ of the document length dampens the effect of document length, preventing severe penalties on extra long documents.

5. Conclusions/Future Research

The use of term weighting in a document produces significant gains in performance, up to a 44% improvement in average precision over simple matching. Additionally the following conclusions can be drawn from the experiments.

a) The three types of measures tested: 1) the importance of a term within a document collection, 2) the importance of a term within a given document, and 3) the length of a document, are all important in term weighting of documents.

b) The two types of term-importance factors, importance within a collection, and importance within a given document, measure term usage in two complementary places--within a given document and within an entire document collection--and combining them produces a cumulative effect.

c) The normalized noise measure of term importance within a document collection is a viable alternative measure to the inverted document frequency measure.

d) Using the $\log_2$ of the frequency of a term within a document instead of its raw frequency produces a superior measure of the importance of a term within a given document.

e) Combining the two types of term importance factors, term importance within a document collection and term importance within a given document, is more effective than using single factors alone. Combining them multiplicatively produces the best results for the test collection and indexing methods used in these experiments.

f) Adding the number of matches between a document and a query to a term weight

produced by any combination involving a factor that measures term importance within a collection does not produce significant improvement in performance, at least for this test collection and for full word indexing.

g) It is important to consider the length of a document in ranking. Dividing the total term weight by the $\log_2$ of the document length produces significant performance improvement for this test collection.

Future research is needed to resolve some of the issues. The $\log_2$ of the frequency and of the document length is only one possible function of these measures, and other moderating functions should be examined. Similarly the normalization of the noise measure should be done using alternative methods.

Additionally these experiments need to be run on other test collections, not only as verification, but to begin to investigate how factor importance varies between collections. This includes the investigation of both single factors, and the combining and weighting of these factors. Various indexing methods, such as suffixing, need to be used, with experimental work done into what changes are needed in document term weighting to best match a given indexing method. A thorough understanding of the role term weighting plays in a statistically-based keyword retrieval system is important for later more advanced research into intelligent user interfaces.

Acknowledgements

Thanks to the rest of the IRX project team, Dennis Benson, Larry Fitzpatrick, Charles Goldstein, and Rand Huntzinger, without whom there would be no research system. Also thanks to the SMART project, especially Chris Buckley, for the use of test collections and system code.

References

[CROFT79] Croft W.B., Harper D.J., "Using Probabilistic Models of Document Retrieval Without Relevance Information", Journal of Documentation, Vol. 35, No. 4, December 1979, pp. 285-295.

[CROFT82] Croft W.B., "Experiments with Representation in a Document Retrieval System", COIN Technical Report 82-21, May 1982.

[DENNIS64] Dennis S.F., "The Construction of a Thesaurus Automatically from a Sample of Text",Symposium Proceedings, Statistical Association Methods for Mechanized Documentation, 1964. (National Bureau of Standards Miscellaneous Publication 269).

[HARTER75] Harter S.P., "A Probabilistic Approach to Automatic Keyword Indexing", Journal of the American Society for Information Science, Vol. 26, No. 5, October 1973, pp. 280-289.

[MCGILL79] McGill M., Koll M., Noreault T., "An Evaluation of Factors Affecting Document Ranking by Information Retrieval Systems". Report, School of Information Studies, Syracuse University, Syracuse, New York, October 1979.

[RAGHAVEN83] Raghavan V.V., Yu C.T., "Evaluation of the 2-Poisson Model as a Basis for Using Term Frequency Data in Searching", Proceedings of the Sixth Annual International ACM SIGIR Conference on Research and Development in Informtion Retrieval, Washington, D.C. 1983.

[SALTON73] Salton G., Yang C.S., "On the Specification of Term Values in Automatic Indexing", Journal of Documentation, Vol. 29, No. 4, December 1973, pp. 351-372.

[SALTON83] Salton G., McGill M., Introduction to Modern Information Retrieval, McGraw-Hill Book Company, New York, 1983.

[SPARCK72] Sparck Jones K., "A Statistical Interpretation of Term Specificity and Its Application in Retrieval", Journal of Documentation, Vol. 28, No. 1, March 1972, pp. 11-20.

[YU82] Yu C.T., Lam K., Salton G., "Term Weighting in Information Retrieval Using the Term Precision Model", Journal of the American Association for Computing Machinery, Vol. 29, No. 1, January 1982, pp. 152-170.

TABLE 1

Run 1 -- number of matches only
Run 2 -- within document frequency only
Run 3 -- log2 within document frequency only
Run 4 -- inverted document frequency weight only
Run 5 -- normalized noise only

Run	1	2	3	4	5
Recall - Precision					
0.00	0.507	0.416	0.532	0.571	0.581
0.10	0.475	0.374	0.491	0.535	0.542
0.20	0.389	0.306	0.417	0.447	0.460
0.30	0.310	0.234	0.315	0.367	0.371
0.40	0.244	0.204	0.270	0.313	0.321
0.50	0.216	0.175	0.230	0.269	0.282
0.60	0.149	0.119	0.158	0.183	0.188
0.70	0.116	0.082	0.119	0.138	0.140
0.80	0.093	0.067	0.095	0.115	0.116
0.90	0.074	0.051	0.074	0.093	0.092
1.00	0.071	0.047	0.069	0.088	0.087
Average precision for 3 intermediate points					
Precision	0.224	0.176	0.237	0.268	0.277
% Precision Change		-21.5	6.0	19.9	23.9
Norm Recall	0.844	0.847	0.853	0.855	0.856
Precis after 10 docs	0.160	0.135	0.175	0.188	0.186
Precis after 30 docs	0.094	0.081	0.096	0.103	0.106
Recall after 10 docs	0.269	0.223	0.290	0.323	0.328
Recall after 30 docs	0.439	0.379	0.446	0.487	0.501
E, 0.5, 10 docs	0.835	0.862	0.820	0.805	0.806
E, 1.0, 10 docs	0.818	0.849	0.803	0.784	0.785
E, 2.0, 10 docs	0.784	0.821	0.766	0.742	0.741
E, 0.5, 30 docs	0.890	0.906	0.889	0.881	0.877
E, 1.0, 30 docs	0.854	0.874	0.852	0.840	0.835
E, 2.0, 30 docs	0.770	0.803	0.768	0.748	0.741
Number queries	225	225	225	225	225

TABLE 2

Run 1 -- 1*number of matches
Run 2 -- 1*number of matches + 1*within document frequency
Run 3 -- 1*number of matches + 3*within document frequency
Run 4 -- 1*number of matches + 1*log2 within document frequency
Run 5 -- 1*number of matches + 3*log2 within document frequency
Run 6 -- 1*number of matches + 1*inverted document frequency weight
Run 7 -- 1*number of matches + 3*inverted document frequency weight
Run 8 -- 1*number of matches + 1*normalized noise
Run 9 -- 1*number of matches + 3*normalized noise

Run	1	2	3	4	5	6	7	8	9
Recall - Precision									
0.00	0.507	0.469	0.442	0.551	0.538	0.575	0.572	0.586	0.587
0.10	0.475	0.432	0.399	0.513	0.501	0.540	0.536	0.547	0.551
0.20	0.389	0.357	0.333	0.429	0.422	0.453	0.451	0.461	0.467
0.30	0.310	0.275	0.250	0.330	0.329	0.366	0.367	0.373	0.376
0.40	0.244	0.238	0.214	0.279	0.279	0.308	0.311	0.319	0.324
0.50	0.216	0.205	0.182	0.244	0.240	0.263	0.268	0.279	0.281
0.60	0.149	0.143	0.128	0.167	0.163	0.177	0.181	0.188	0.187
0.70	0.116	0.106	0.092	0.127	0.123	0.135	0.135	0.142	0.141
0.80	0.093	0.085	0.073	0.104	0.099	0.111	0.112	0.118	0.117
0.90	0.074	0.066	0.057	0.084	0.077	0.089	0.090	0.095	0.093
1.00	0.071	0.062	0.053	0.079	0.072	0.084	0.085	0.089	0.088
Average precision for 3 intermediate points									
Precision	0.224	0.208	0.189	0.247	0.244	0.265	0.268	0.277	0.278
% Precision Change		-7.0	-15.6	10.2	9.0	18.7	19.8	23.7	24,5
Norm Recall	0.844	0.851	0.849	0.853	0.853	0.855	0.855	0.857	0.857
Precis after 10 docs	0.160	0.153	0.147	0.180	0.180	0.187	0.188	0.191	0.189
Precis after 30 docs	0.094	0.088	0.084	0.097	0.096	0.103	0.103	0.105	0.105
Recall after 10 docs	0.269	0.252	0.239	0.301	0.301	0.322	0.325	0.327	0.333
Recall after 30 docs	0.439	0.419	0.391	0.461	0.453	0.489	0.487	0.493	0.495
E, 0.5, 10 docs	0.835	0.843	0.849	0.814	0.814	0.806	0.805	0.802	0.803
E, 1.0, 10 docs	0.818	0.828	0.836	0.795	0.795	0.786	0.785	0.782	0.782
E, 2.0, 10 docs	0.784	0.796	0.807	0.757	0.757	0.743	0.742	0.739	0.737
E, 0.5, 30 docs	0.890	0.898	0.903	0.887	0.888	0.880	0.881	0.878	0.878
E, 1.0, 30 docs	0.854	0.864	0.871	0.849	0.851	0.840	0.840	0.837	0.837
E, 2.0, 30 docs	0.770	0.785	0.798	0.762	0.766	0.747	0.748	0.744	0.744
Number queries	225	225	225	225	225	225	225	225	225

TABLE 3

Run 1 -- number of matches
Run 2 -- 1*log2 within document frequency
Run 3 -- 1*inverted document frequency weight
Run 4 -- 1*normalized noise
Run 5 -- 1*log2 within document frequency + 1*inverted document frequency weight
Run 6 -- 1*log2 within document frequency + 1*normalized noise
Run 7 -- 1*log2 within document frequency + 3*normalized noise
Run 8 -- 3*log2 within document frequency + 1*normalized noise
Run 9 -- 1*log2 within document frequency * 1*inverted document frequency weight
Run 10-- 1*log2 within document frequency * 1*normalized noise

Run	1	2	3	4	5	6	7	8	9	10
Recall - Precision										
0.00	0.507	0.532	0.571	0.581	0.579	0.594	0.592	0.582	0.594	0.607
0.10	0.475	0.491	0.535	0.542	0.546	0.555	0.555	0.544	0.558	0.575
0.20	0.389	0.417	0.447	0.460	0.468	0.484	0.483	0.464	0.476	0.494
0.30	0.310	0.315	0.367	0.371	0.376	0.388	0.389	0.369	0.387	0.397
0.40	0.244	0.270	0.313	0.321	0.322	0.337	0.335	0.316	0.341	0.354
0.50	0.216	0.230	0.269	0.282	0.275	0.290	0.289	0.269	0.291	0.305
0.60	0.149	0.158	0.183	0.188	0.195	0.205	0.201	0.189	0.201	0.215
0.70	0.116	0.119	0.138	0.140	0.147	0.155	0.153	0.144	0.148	0.156
0.80	0.093	0.095	0.115	0.116	0.122	0.129	0.125	0.119	0.121	0.128
0.90	0.074	0.074	0.093	0.092	0.097	0.103	0.100	0.094	0.093	0.097
1.00	0.071	0.069	0.088	0.087	0.092	0.097	0.094	0.088	0.086	0.091
Average precision for 3 intermediate points										
Precision	0.224	0.237	0.268	0.277	0.277	0.290	0.290	0.275	0.288	0.301
% Precision Change		6.0	19.9	23.9	23.8	29.8	29.5	23.1	28.9	34.7
Norm Recall	0.844	0.853	0.855	0.856	0.859	0.861	0.859	0.860	0.862	0.863
Precis after 10 docs	0.160	0.175	0.188	0.186	0.199	0.201	0.195	0.198	0.203	0.210
Precis after 30 docs	0.094	0.096	0.103	0.106	0.107	0.109	0.109	0.105	0.108	0.111
Recall after 10 docs	0.269	0.290	0.323	0.328	0.336	0.342	0.338	0.339	0.341	0.356
Recall after 30 docs	0.439	0.446	0.487	0.501	0.500	0.510	0.514	0.495	0.508	0.519
E, 0.5, 10 docs	0.835	0.820	0.805	0.806	0.795	0.792	0.798	0.795	0.791	0.783
E, 1.0, 10 docs	0.818	0.803	0.784	0.785	0.774	0.771	0.776	0.774	0.771	0.762
E, 2.0, 10 docs	0.784	0.766	0.742	0.741	0.731	0.727	0.732	0.731	0.728	0.717
E, 0.5, 30 docs	0.890	0.889	0.881	0.877	0.876	0.873	0.873	0.878	0.875	0.871
E, 1.0, 30 docs	0.854	0.852	0.840	0.835	0.835	0.831	0.831	0.838	0.833	0.829
E, 2.0, 30 docs	0.770	0.768	0.748	0.741	0.740	0.734	0.733	0.744	0.737	0.731
Number queries	225	225	225	225	225	225	225	225	225	225

TABLE 4

Run 1 -- number of matches only
Run 2 -- log2 within document frequency * normalized noise
Run 3 -- (log2 within document frequency * normalized noise)/document length
Run 4 -- (log2 within document frequency + inverted document frequency weight)/log2 document length
Run 5 -- (log2 within document frequency * inverted document frequency weight)/log2 document length
Run 6 -- (log2 within document frequehcy + normalized noise)/log2 document length
Run 7 -- (log2 within document frequency * normalized noise)/log2 document length

Run	1	2	3	4	5	6	7
Recall - Precision							
0.00	0.507	0.607	0.561	0.622	0.631	0.631	0.629
0.10	0.475	0.575	0.523	0.588	0.607	0.602	0.606
0.20	0.389	0.494	0.459	0.510	0.520	0.526	0.528
0.30	0.310	0.397	0.388	0.417	0.418	0.433	0.428
0.40	0.244	0.354	0.325	0.346	0.364	0.366	0.372
0.50	0.216	0.305	0.275	0.297	0.314	0.308	0.326
0.60	0.149	0.215	0.201	0.207	0.227	0.219	0.234
0.70	0.116	0.156	0.169	0.160	0.168	0.171	0.172
0.80	0.093	0.128	0.141	0.132	0.139	0.142	0.142
0.90	0.074	0.097	0.108	0.106	0.108	0.112	0.108
1.00	0.071	0.091	0.101	0.100	0.102	0.106	0.101
Average precision for 3 intermediate points							
Precision	0.224	0.301	0.287	0.305	0.316	0.318	0.322
% Precision Change		34.7	28.3	36.2	41.0	41.9	44.0
Norm Recall	0.844	0.863	0.868	0.862	0.866	0.864	0.866
Precis after 10 docs	0.160	0.210	0.198	0.212	0.224	0.217	0.228
Precis after 30 docs	0.094	0.111	0.109	0.112	0.114	0.113	0.115
Recall after 10 docs	0.269	0.356	0.337	0.356	0.376	0.364	0.382
Recall after 30 docs	0.439	0.519	0.508	0.527	0.532	0.529	0.535
E, 0.5, 10 docs	0.835	0.783	0.795	0.781	0.770	0.776	0.765
E, 1.0, 10 docs	0.818	0.762	0.774	0.759	0.748	0.754	0.743
E, 2.0, 10 docs	0.784	0.717	0.730	0.714	0.700	0.708	0.695
E, 0.5, 30 docs	0.890	0.871	0.873	0.870	0.867	0.869	0.866
E, 1.0, 30 docs	0.854	0.829	0.831	0.827	0.823	0.825	0.822
E, 2.0, 30 docs	0.770	0.731	0.735	0.727	0.723	0.725	0.721
Number queries	225	225	225	225	225	225	225