Report on the TREC-4 Experiment: Combining Probabilistic and Vector-Space Schemes

Jacques Savoy, Melchior Ndarugendamwo, Dana Vrajitoru

Faculté de droit et des sciences économiques
Université de Neuchâtel
Pierre-à-Mazel 7
CH - 2000 Neuchâtel (Switzerland)

E-mail: Jacques.Savoy@unine.ch
Web page: www.unine.ch/info/

in Proceedings TREC'4, October 1996, NIST Publication 500-236, p. 537-548

Summary

This paper describes and evaluates a retrieval scheme combining the OKAPI probabilistic retrieval model with various vector-space schemes. In this study, each retrieval strategy represents both queries and documents using the same set of single terms; however they weight them differently. To combine these search schemes, we do not apply a given combination operator on the retrieval status nor the rank of each retrieved record (e.g., sum, average, max., etc.). We think that each retrieval strategy may perform well for a set of queries and poorly for other requests. Thus, based on a given query's statistical characteristics, our search model first selects the more appropriate retrieval scheme and then retrieves information based on the selected search mechanism. Since the selection procedure is done before any search operation, our approach has the advantage of limiting the search time to one retrieval algorithm instead of retrieving items using various retrieval schemes, and then combining the given results.

In particular, this study addresses the following questions: (1) can the statistical characteristics of a query be good predictors in an automatic selection procedure; (2) faced with the relatively high retrieval effectiveness achieved by the OKAPI model, can various vector-space schemes further improve the retrieval performance of the OKAPI approach, and (3) can the learning results obtained with one tested collection (WSJ) be valid for another corpus (SJMN)?

Participation: Category: B Query: ad-hoc, fully automatic

Introduction

There are many reasons for using multiple sources of evidence [Katzer et al. 1982] [Tenopir 1985]. Firstly, the studies comparing the same document representation scheme do not always produce the same result, because they are not based on the same domain of knowledge, they use different collections of documents, different stemming algorithms, etc. Secondly, when we compare the performance (e.g., recall and precision) of different representations, none is found to perform well for all criteria. Thirdly, a comparison of mean performances among various search strategies reveals that when a difference occurs, it is small. Fourthly, even for representations considered similar, such as "abstract" and "title and abstract", the overlap between pairs of representations is very low (around 35%); therefore, one cannot assert that distinct document representations can be considered as equivalent (see also [Noreault et al. 1981] about representations built on controlled vs. free-text vocabularies). Fifthly, when studying the overlap in retrieved items by various searchers searching the same question, Saracevic & Kantor [1988, p. 204-207] find that the intersection is relatively low (e.g., for around 78.7% of all items retrieved, the degree of agreement is less than 25%). This fact cannot be explained by a low search term overlap however:

> "the conclusion that different searchers for the same question see and interpret different things in a question, represent them by different linguistic and/or logical constructs, and retrieve different things from a file." [Saracevic & Kantor, 1988, p. 204]

Finally, the analysis of various TREC experiments [Harman 1994] demonstrates that a given retrieval scheme may perform very well for some queries and poorly for other ones. Therefore, overall statistics, like the average precision, may hide performance irregularities among requests when comparing different retrieval schemes. To overcome these problems, the combination of retrieval schemes seems to be a necessity.

The integration of multiple sources of information (especially provided by other retrieval information schemes) is currently analyzed in two different contexts. These retrieval strategies may operate on the same collection (data fusion problem), on the one hand, and, on the other hand, they may retrieve items for a given request from different corpora or various information servers (collection fusion problem). This latter question involves the merging the retrieval results of searches on independent collections into an effective, single ranked list [Voorhees et al. 1995, April], [Voorhees et al. 1995, July], and particularly, this problem appears in distributed systems, such as the WWW.

In this study, we are concerned with the data fusion problem, where an important question does arise : is it pertinent to consider multiple retrieval schemes operating on the same collection? The answer seems to be positive. For example, Saracevic & Kantor [1988, p. 204-207] demonstrate that the odds of a document being relevant to a given request increases monotonically with the number of search strategies that retrieve this record. Moreover, for items retrieved only once, the relevance odds decrease about 8 to 10. Other studies reveal that one can increase the performance of a retrieval system by using multiple document surrogates, various query formulations or by combining multiple search schemes (e.g., [Fox et al. 1988], [Turtle & Croft 1991], [Thompson 1993], [Fox et al. 1993], [Belkin et al. 1993], [Belkin et al. 1994], [Bartell et al. 1994], [Fox & Shaw 1994], [Shaw & Fox 1995], [Lee 1995]).

Traditionally, in studying the data fusion problem, the retrieval engine first find the retrieved set of each retrieval scheme and then defines an appropriate merging function. For this combination, we may consider the rank of the retrieved records and / or their retrieval status value. In this vein, Fox et al. [1993], [Fox & Shaw 1994], [Shaw & Fox 1995] show various ad hoc schemes in combining the p-norm model and vector-processing strategies. Of course, these retrieval schemes may be based on different indexing strategies of the same collection. In this latter case, the retrieval status values may not have a range of possible similar values, leading to a more complex combination problem (see also [Lee 1995]). Moreover, we may need to weight each retrieval scheme (or define a predicting relevance based on the relative merit of each single search strategy) based on previous relevance assessments.

In this study, we do not consider really different requests (e.g., Boolean and natural language queries) or document representations (e.g., single terms vs. phrase indexing strategies). Each query or document is represented by the same set of single terms and the weight assigned to each keyword may vary according to each retrieval scheme. Since our selection procedure takes place before any retrieval operation, our retrieval cost is limited to only one search algorithm instead of summing (at least with von Neuman architecture), the computation time required by each single search scheme [Fox et al. 1993], [Fox & Shaw 1994], [Shaw & Fox 1995], [Lee 1995].

Our retrieval procedure can be viewed as a trial and error process [Swanson 1977]. Moreover, in this spirit, if our automatic selection procedure fails for a given request to choose the best retrieval scheme, the user may have the opportunity to select a more appropriate search strategy (at least, in the user's opinion).

The rest of this paper is organized as follows. The next section presents an overview of both the vector-space and the OKAPI probabilistic models. The second section describes the basic principles of the k-Nearest-Neighbors method (k-NN) used to select the more appropriate retrieval scheme based on statistical features of each query.

1. Retrieval Models

To define a retrieval model, we must explain how documents and queries are represented and how these representations are compared to produce a ranked list of retrieved items. In this experiment, the indexing procedure done by the SMART system is fully automatic and based on a single term only.

To achieve this goal, each topic was indexed according to the content of its Descriptive (<desc>) section only. For each document, Text (<text>) section as well as Subtitle (<ST>), Headline (<HL>), and Summary (<LP>) sections were indexed for the WSJ collection (<leadpara>, <text> and <headline> for the SJMN corpus). All other subsections were removed, and, in particular, the title, the narrative and the concept section of each topic (see Table 1).

Collection	Section
WSJ	<desc>, <text>, <st>, <hl>, <lp>
SJMN	<leadpara>, <text>, <headline>
Query	<desc>

Table 1: Selected Sections Used to Represent Documents and Queries

	Indexing Weight
NNN	$w_{ij} = tf_{ij}$
ANN	$w_{ij} = 0.5 + 0.5 \cdot \frac{tf_{ij}}{\max tf_{ik}}$
LNC	$w_{ij} = \frac{\log(tf_{ij})+1}{\sqrt{\sum_{k=1}^{t} (\log(tf_{ik})+1)^2}}$
LTC	$w_{ij} = \frac{[\log(tf_{ij})+1] \cdot idf_j}{\sqrt{\sum_{k=1}^{t} ([\log(tf_{ik})+1] \cdot idf_k)^2}}$

Table 2: Indexing Weighting Schemes

As shown in Section 1.1, various weighting schemes can be used within the vector-space model, leading to different retrieval effectiveness. Moreover, this section demonstrates that the length of the query may play an important role in the retrieval performance. Section 1.2 describes the probabilistic model OKAPI based on a different weighting and matching strategy.

1.1. Evaluation of the Vector Space Model

To represent each document and each query by a vector of weighted keywords, the vector-space model suggests various weighting schemes. To select the more appropriate one, we have conducted a set of experiments based on different weighting formulas. To assign an indexing weight w_{ij} reflecting the importance of each single-term T_j, j = 1, 2, ..., t, in a document D_i, we may use one of the equations shown in Table 2. In this table, tf_{ij} depicts the frequency of the term T_j in the document D_i (or in the request), n represents the number of documents D_i in the collection, df_j the number of documents in which T_j occurs, and idf_j the inverse document frequency (log $[n / df_j]$).

To normalize each indexing weight between 0 and 1, we may consider the cosine normalization (see LNC formula in Table 2), or we may also take account of the distribution of each indexing term in the collection by giving a higher weight to sparse words and lower importance to more frequent terms (idf component in LTC formula in Table 2).

To define the retrieval status value (RSV) of each document D_i, the vector-space model uses the following equation:

$$RSV_{VSM}(D_i) = \sum_{k=1}^{q} w_{ik} \cdot w_{qk} \qquad (1)$$

where w_{ik} represents the indexing term weight of T_k in a document D_i, w_{qk} the keyword search weight of T_k in the current query and q the number of search keyword in the request.

In Table 3, we compare the retrieval effectiveness achieved using various indexing schemes and three different query formulations. In this table and in the following tables, precision is measured at eleven standard recall values (from 0.0 to 1.0) for all queries (#1 through #200), and then averaged to form our retrieval effectiveness measure. The numbers in parenthesis indicate the percent of change computed by the system based on the baseline solution.

Model \ Query Form	Average Precision (% change)		
	<desc> (baseline)	<desc> and <narr>	<desc>, <narr> and <title>
Vector-Space Model			
doc = NNN, query = NNN	4.59	7.03 (+53.2)	9.42 (+105.2)
doc = ANN, query = ANN	9.41	10.22 (+8.6)	13.07 (+38.9)
doc = LNC, query = LNC	9.64	19.79 (+105.3)	22.48 (+133.2)
doc = LTC, query = LTC	15.69	23.26 (+48.2)	25.63 (+63.35)
doc = LTC, query = LNC	12.10	21.11 (+74.5)	23.58 (+94.9)
doc = LNC, query = LTC	17.04	27.42 (+60.8)	29.95 (+75.8)
Probabilistic Retrieval Model			
OKAPI	22.56	33.01 (+46.3)	32.49 (+44.0)

Table 3: Evaluation of Individual Retrieval Schemes (WSJ Collection)

To decide whether a search strategy is better than another, a difference of at least 5% in average precision is generally considered significant and, a 10% difference is considered very significant.

For a long query, the Descriptive, Narrative and Title sections of the topic description were used to build the request vector, while the shortest query form is built only with the Descriptive section (the new request form for TREC'4). The middle column shows the retrieval performance achieved by using both the Descriptive and the Narrative sections.

From the data shown in Table 3, we may find that:

1. for all request representations, the OKAPI probabilistic model achieves the highest retrieval performance;
2. when considering only the vector-space model, the best result is obtained when using the LNC-LTC strategy.
3. when the request representation includes more information about the user's information need, the retrieval performance is enhanced. The only exception to this rule is the OKAPI model which achieved similar performance using a long or a medium size query formulation (33.01 vs. 32.49 (-1.6%)).

1.2. OKAPI Probabilistic Model

Based on TREC'3 results [Robertson et al. 1995b], the OKAPI probabilistic approach presents a very attractive retrieval model. This model is based on the combination of two probabilistic schemes using information about term frequency both in the request and in the document, and incorporating a correction factor to account for document length.

The OKAPI probabilistic model is based on: (1) the weighting of the search term as a traditional probabilistic model (represented by the component $w^{(1)}$); (2) the frequency of the indexing term (component tf_{ik}); (3) the frequency of the search term (component tf_{qk}), and (4) a length correction factor (component avdl) taking account of document surrogate length. Schematically, the computation of the retrieval status value of each document is expressed as:

$$RSV_{BM}(D_i) = \text{CorrectionFactor} + \sum_{k=1}^{q} w_{ik} \cdot w_{qk}$$

A formal derivation of w_{qk} assigned to each search keyword is obtained in the probabilistic retrieval model by making use of Bayes' theorem and term-independence assumption postulating that the index terms occur independently in the relevant and nonrelevant document (for details see [van Rijsbergen 1979, Chapter 6]). In this case, the weight w_{qk} is evaluated according to Formula 2.

$$w^{(1)} = \log\left[\frac{r_{qk}}{1 - r_{qk}}\right] + \log\left[\frac{1 - s_{qk}}{s_{qk}}\right] = \log\left[\frac{n - df_k}{df_k}\right] + c \qquad (2)$$

$$\text{with } c = \log\left[\frac{r_{qk}}{1 - r_{qk}}\right]$$

in which r_{qk} (s_{qk}) expresses the conditional probability of knowing the document is relevant (nonrelevant), its representative containing the index term T_k.

The combined approach, called BM25, is based on the following evaluation of the retrieval status value:

Parameter	$BM25_{WSJ}$	$BM25_{CACM}$	$BM25_{CISI}$
k_1 =	2	2	2
k_2 =	0	0	0
k_3 =	∞	5	∞
b =	0.75	0.375	0.125
c =	1	1	1
s_1 =	2	2	2
s_3 =	1	5	1
r_{qk} =	0.5	0.6	0.5
avdl =	241.332	33.9919	67.6062
n =	173,252	3,204	1,460

Table 4: Parameter Setting for OKAPI Probabilistic Model

$$RSV_{BM25}(D_i) = k_2 \cdot l_q \cdot \frac{avdl - l_i}{avdl + l_i} + \sum_{k=1}^{q} s_1 \cdot \frac{tf_{ik}^{c}}{K^{c} + tf_{ik}^{c}} \cdot w^{(1)} \cdot s_3 \cdot \frac{tf_{qk}}{k_3 + tf_{qk}} \qquad (3)$$

$$\text{with } K = k_1 \cdot \left[(1 - b) + b \cdot \frac{l_i}{avdl}\right]$$

where avdl means the average document surrogate length, l_i (l_q) represents the document representative length (query length, respectively), k_1, k_2, k_3, s_1, s_3 are unknown constants, and $w^{(1)}$ is estimated by Equation 2.

In order to define an "optimal" parameter setting for the BM25 model, we have to conduct a set of experiments based on the CACM and CISI test-collections [Savoy 1995]. The results are depicted in Table 4. However, in our current context, we have set our retrieval scheme according to the parameter values given by [Robertson et al. 1995b] (or $BM25_{WSJ}$ in Table 4).

2. Combination of Various Retrieval Models

In the previous section, we described two retrieval models that can be used to retrieve information from a textual database. However, it is known that a given search scheme may perform well for some requests but poorly for other ones. This phenomenon is clearly demonstrated by the analysis of variance described in [Tague-Sutcliffe & Blustein 1995] indicating that the variability attributable to the queries is much greater than those due to the different search strategies. However, in referring to the retrieval effectiveness shown in Table 3, we might ask whether we can really improve the particularly high performance achieved by the OKAPI probabilistic model.

To answer this question, we have designed a selection procedure which, based on query features, automatically selects the "best" single retrieval strategy for the current request. This problem can be viewed as a classification task within which a decision has to be made on the basis of currently available information. Various learning schemes can be conceived, such as Bayesian classifiers (linear or quadratic discriminant functions), k-Nearest-Neighbors (k-NN), Neural Networks, logistic regression, rule-based methods or Decision Trees [Weiss & Kulikowski 1991], [Michie et al. 1994].

For our purposes, we have chosen the k-NN method already used in IR studies in different contexts; for example, to define the number of documents to be selected from different information servers [Voorhees et al. 1995, April]. In this case, the similarity between queries is computed based on the keywords contained in the different requests and not based on their statistical characteristics. Moreover, Michie et al. [1994, p. 185] have demonstrated that the k-Nearest-Neighbors method performs similar to statistical methods which perform particularly well when the selected features are good predictors, and seem to perform better than rule-based or Decision Trees methods.

2.1. Is a Selection Procedure Really Useful?

From previous research, we can conclude that the combination of various retrieval schemes is a useful strategy for enhancing retrieval effectiveness. However, in our context, we do not really combine the retrieved items from different retrieval schemes, but our system tries to select a single retrieval based on the statistical characteristics of the current request. For a given request, this strategy has the advantage that only one single retrieval mechanism has to be computed.

Model	Average Precision	# best run	# best run $\delta = 2\%$	# best run $\delta = 5\%$
Vector-Space Model				
1. doc = NNN, query = NNN	4.59	10	10	10
2. doc = ANN, query = ANN	9.41	13	13	12
3. doc = LNC, query = LNC	9.64	7	7	7
4. doc = LTC, query = LTC	15.69	24	22	21
5. doc = LTC, query = LNC	12.10	2	1	1
6. doc = LNC, query = LTC	17.04	12	12	12
Probabilistic Retrieval Model				
7. OKAPI	22.56	132	135	137
Combined Approach				
Using schemes 1, 2, 3, 4, 5, 6, 7		24.10	24.09	24.09

Table 5: Characteristics of Individual Retrieval Schemes (WSJ Collection)

However, our selection procedure must choose between one probabilistic retrieval strategy having a particularly high mean retrieval effectiveness and six vector-space schemes as shown in Table 5. Based on 200 queries, Table 5 depicts the average precision and, for each retrieval scheme, the number of best individual runs on a per query basis. Thus, for 132 queries out of 200, the best choice is the OKAPI model. It is interesting to note that the best vector-space approach (document = LNC, query = LTC) does not result in a significantly different number of best runs compared to other vector-space schemes.

From Table 5, one can see that:

1. the overall good performance of the OKAPI model hides some irregularities;
2. even a simple retrieval scheme like the vector-space model based on the NNN or ANN indexing scheme represents the best scheme for 13 requests out of 200 (or 11.5% of the cases);
3. the best vector-space scheme (document = LNC, query = LTC) does not represent the highest number of best runs for the vector-space model;
4. the optimal selection may enhance the OKAPI probabilistic retrieval model of around 7% (24.10 vs. 22.56).

However, one can argue that when a vector-space model reveals a better retrieval performance, the difference must be small compared to the OKAPI model. In response to this question, Table 5 depicts the number of best individual runs on a per query basis, provided that the difference δ is greater than 2% (or 5%) compared to the OKAPI model. The data shown in this table is clear: when a difference occurs, it is greater than 5% compared to the probabilistic retrieval strategy, or in other words, ignoring small differences it leads to a similar retrieval performance (24.09 vs. 24.10).

2.2. Principles of k-Nearest-Neighbors (k-NN)

The aim of our selection problem can be stated as follows (see Table 6):

- given a set of search scheme_h, h = 1, 2, ..., q;
- given a set of query features f_i, i = 1, 2, ..., p;
- given a set of query Q_j, j = 1, 2, ..., m for which we know both the values of each feature and the best retrieval scheme;
- find the optimum retrieval scheme_i for a new request Q knowing the values of each of its features.

As described in the previous section, this study evaluates seven retrieval schemes. As query features, other studies have considered the number of terms in the request, the sum of the idf over all search keywords, the mean of these query search weights [Croft & Thompson 1984], [Croft & Thompson 1987]. In this study, we have retained the following seven statistical features:

1. the number of search included in the request;
2. the maximum term frequency (tf);
3. the tf mean;
4. the maximum inverse document frequency (idf);
5. the idf min;
6. the idf mean;
7. the idf median.

This list does not include the term frequency median (tf median) because this measure has a null standard deviation. Finally, based on this data, we must select a classification method to predict the retrieval scheme to apply to each new request. In this study, we have chosen a simple learning method, the k-NN method which works as follows.

Query	Features Schemes				Search
Q1	x11	x12	...	x1p	scheme1
Q2	x21	x22	...	x2p	schemeq
...	...	...	...	...	...
Qj	xj1	xj2	...	xjp	scheme2
...	...	...	...	...	...
Qm	xm1	xm2	...	xmp	scheme1
New Q	x_1	x_2	...	x_p	?

Table 6: Example of a Classification Task

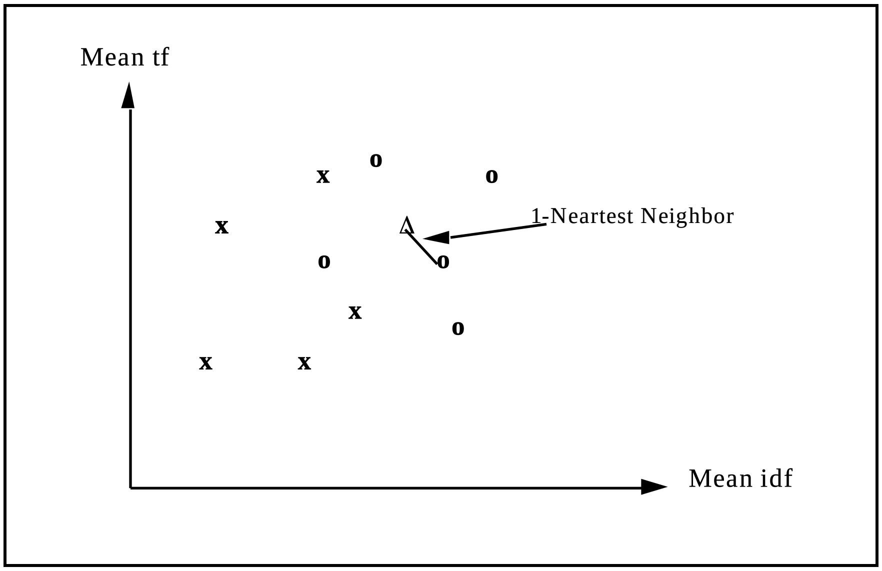

Figure 1: The Nearest Neighbor (1-NN) of a New Request

For each new topic Q, the system must find the k nearest neighbors in the set of all existing request Q_j, j = 1, 2, ..., m. To define a suitable metric for this purpose, the Euclidean distance has been chosen. More precisely, the difference between the values of each feature is squared and summed over all features. The square root of this sum defines the actual Euclidean distance and the minimum distance indicates the nearest neighbor.

For example, in Figure 1, an "x" indicates each query for which the best retrieval scheme is A, and "0" each request for which the best search strategy is B. The new query is depicted by "Δ". After computing the Euclidean distance from this new observation to all others, we find that the closest distance is a query noted "0", leading to the prediction that the best retrieval scheme for this new query is the search strategy B. However, the real situation is more complex as shown in Figure 2.

In the k-NN method, if the constant k is defined as one, the selected search technique is simply the nearest neighbor of the new request (see Figure 1). For another value of k (e.g., five in this study), the system selects the retrieval schemes appearing most often in the set of the k closest requests, and ties, if any, are broken using the prior probability (see Section 2.3).

A final implementation isuue should be discussed. Since each query feature is measured with a different scale, we have standardized each feature value by subtracting the estimated mean and dividing by the estimated standard error. Thus the distance between two observations is computed in terms of standard deviation from the sample mean of each feature.

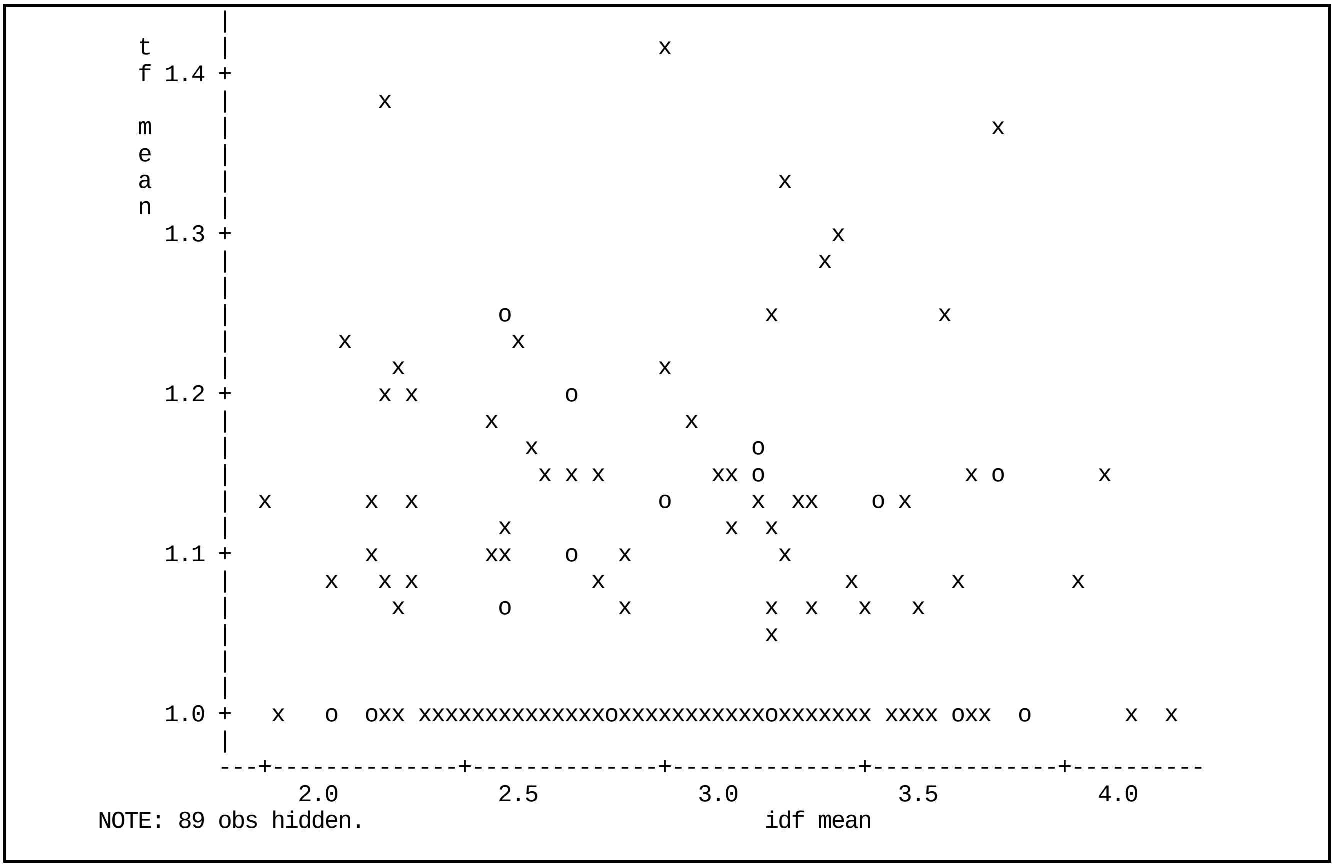

Figure 2: Scattergram of Two Query Characteristics (tf Mean, idf Mean) According to Two Retrieval Schemes

With each classification method, we must deal with various problems. Firstly, adding more features does not necessary lead to a better prediction results. Secondly, some chosen features may reveal little discrimination power between each search scheme. Thirdly, some features may be redundant to others characteristics. Fourthly, in our attempt to learn from the data, we may eventually infer that better features are needed to make reliable predictions. Finally, we implicitly admit that the sample of queries used to built our selection procedure is a representative sample of future requests.

In a related study, McCall & Willett [1986] suggest taking account of the query similarity related to the number of documents within which the search terms occurs (Dice's coefficient) on the one hand, and, on the other hand, the mean similarity between the request and the top 10 retrieved records of each search strategies. Finally, these authors suggest considering the number of relevant items retrieved by each search mechanisms.

> "In view of these results and those obtained by Croft & Thompson [1984], we think it unlikely that automatic selection criteria will be sufficiently discriminating to choose between different search mechanisms in multi-strategy retrieval systems." [McCall & Willett 1986, p. 325]

A direct comparison with this study is not possible because our selection approach must be performed before any search operation. Therefore, in our context, we ignore both the retrieved set and the relevant records obtained by each search scheme.

2.3. The Default Rule

As a first attempt to define a selection rule, we may ignore the query features and based our decision on the prior probabilities. The resulting decision, called the default rule (no data), of our selection procedure is the following: "Select the retrieval scheme having the maximum prior probability". Based on data of Table 5, we may conclude that the OKAPI model returns the best retrieval performance for 132 requests out of 200 (or for 66% of the queries). Thus, the selection of this probabilistic retrieval model is always the response of the default rule.

The evaluation under the label "UNINE3" reflects the result achieved by the OKAPI probabilistic retrieval model for queries from #202 to #250 and represents the baseline approach used for comparisons.

2.3. Official Runs

Considering all the statistical characteristics of each query, our learning scheme based on the 5-NN approach suggests that all queries from #202 to #250 must be processed according to the OKAPI search model. However, this solution is the same as the default rule leading to the conclusion that the picked statistical features of queries cannot be considered as good predictors for an automatic selection of the search strategy.

However, if the computation of the nearest neighbor of a query is based on (1) the number of search keywords, (2) the maximum of the search term frequency, (3) the mean of the search term frequency, (4) the maximum inverse document frequency and (5) the median of the inverse document frequency, another selection will be produced. Thus, the evaluation under the label "UNINE4" reflects the result achieved by our learning procedure based on these five request features (see Table 7).

The selection result depicted in Table 7 is obtained through a learning process based only on the WSJ collection and queries #1 to #200. However, the following question must still be answered: is this selection the optimum for the WSJ collection and for the SJMN corpus when considering topics #202 to #250 (without topic #236)?

When comparing the retrieval effectiveness of the default rule (average precision of 17.97 - 48 queries) with our selection paradigm (17.14), we can see that our approach decreases the overall performance by around 4.6%. Even if this difference cannot be considered as significant, we must still investigate whether this difference appears only on the SJMN collection or also with the WSJ collection (on which the learning process has been done). In Table 8, one can find the optimum retrieval scheme for each query; this optimum selection leads to an average precision of 21.26.

Conclusion

Before all the needed statistical analyses are performed, our preliminary feelings are the following. We think that all statistical query features are not good predictors for an automatic selection procedure. The selection of the appropriate retrieval scheme based on request features is also a difficult problem because the underlying characteristics do not possess very effective powers of discrimination.

Moreover, the optimal combination of various vector-space schemes does not very significantly enhance the particularly high retrieval effectiveness of the OKAPI model. Thus, faced with a good search strategy, an appropriate combination of retrieval schemes seems to be grounded on a radically different indexing strategy (e.g., based on phrase [Bartell et al. 1994]) or on different query formulations (e.g., p-norm and natural language requests [Fox & Shaw 1994]).

However, when considering computation costs, the selection of an appropriate search scheme before any retrieval operation takes place represents an interesting approach.

Query	Search Scheme
204	doc = NNN, query = NNN
233	doc = ANN, query = ANN
205, 238	doc = LTC, query = LTC
others	doc = OKAPI, query = OKAPI

Table 7: Selection of the "Appropriate" Retrieval Scheme (Based on the WSJ Collection)

Query	Search Scheme
	doc = NNN, query = NNN
208, 214, 230, 232, 249	doc = ANN, query = ANN
210, 238	doc = LNC, query = LNC
216, 217, 229, 239, 240	doc = LTC, query = LTC
206, 212	doc = LTC, query = LNC
204, 213, 228, 241	doc = LNC, query = LTC
others	doc = OKAPI, query = OKAPI

Table 8: Selection of the "Optimum" Retrieval Scheme

Acknowledgments

The authors would like to thank Prof. G. Salton and C. Buckley from Cornell University for giving us the opportunity to use the SMART system without which this study could not be conducted. This research was supported by the SNFSR (Swiss National Foundation for Scientific Research) under grants 2100-037345.93 and 20-43'217.95.

References

[Bartell 1994] B. T. Bartell, G. W. Cottrell, R. K. Belew: Automatic Combination of Multiple Ranked Retrieval Systems. Proceedings ACM-SIGIR'94, Dublin (Ireland), June 1994, 173-181.

[Belkin 1993] N. J. Belkin, C. Cool, W. B. Croft, J. P. Callan: The Effect of Multiple Query Representations on Information System Performance. Proceedings ACM-SIGIR'93, Pittsburgh (PA), June 1993, 339-346.

[Belkin 1994] N. J. Belkin, P. Kantor, C. Cool, R. Quatrain: Query Combination and Data Fusion for Information Retrieval. Proceedings TREC'2, Gaithersburg (MD), April 1994, 35-44.

[Buckley 1995] C. Buckley, G. Salton, J. Allen, A. Singhal: Automatic Query Expansion using SMART. Proceedings TREC'3, Gaithersburg (MD), April 1995, 69-80.

[Callan 1995] J. P. Callan, Z. Lu, W. B. Croft: Searching Distributed Collections with Inference Networks. Proceedings ACM-SIGIR'95, Seattle (WA), July 1995, 21-28.

]Croft 1984] W. B. Croft, R. H. Thompson: The Use of Adaptive Mechanisms for Selection of Search Strategies in Document Retrieval Systems. Proceedings ACM-SIGIR'84, Cambridge (UK), July 1984, 95-110.

[Croft 1987] W. B. Croft, R. H. Thompson: I^3R: A New Approach to the Design of Document Retrieval Systems. Journal of the American Society for Information Science, 38(6), 1987, 389-404.

[Fox 1988] E. A. Fox, G. L. Nunn, W. C. Lee: Coefficients for Combining Concept Classes in a Collection. Proceedings ACM-SIGIR'88, Grenoble (France), June 1988, 291-307.

[Fox 1993] E. A. Fox, M. P. Koushik, J. Shaw, R. Modlin, D. Rao: Combining Evidence from Multiple Searches. Proceedings TREC'1, Gaithersburg (MD), March 1993, 319-328.

[Fox 1994] E. A. Fox, J. A. Shaw: Combination of Multiple Searches. Proceedings TREC'2, Gaithersburg (MD), March 1994, 243-249.

[Harman 1994] D. Harman (Ed.): Proceedings TREC'2. Gaithersburg (MD), April 94.

[Katzer 1982] J. Katzer, M. J. McGill, J. A. Tessier, W. Frakes, P. DasGupta: A Study of the Overlap among Document Representations. Information Technology: Research & Development, 2, 1982, 261-274.

[Lee 1995] J. H. Lee: Combining Multiple Evidence from Different Properties of Weighting Schemes. Proceedings ACM-SIGIR'95, Seattle (WA), July 1995, 180-188.

[McCall 1986] F. M. McCall, P. Willett: Criteria for the Selection of Search Strategies in Best Match Document Retrieval Systems. International Journal of Man-Machine Studies, 25(3), 1986, 317-326.

[Michie 1994] D. Michie, D. J. Spiegelhalter, C. C. Taylor(Ed.): Machine Learning, Neural and Statistical Classification. Ellis Horwood, New-York (NY), 1994.

[Noreault 1981] T. Noreault, M. J. McGill, M. B. Koll: A Performance Evaluation of Similarity Measures, Document Term Weighting Schemes and Representations in a Boolean Environment. In Information Retrieval Research, R.N Oddy, S. E. Robertson, c. J. van Rijsbergen, P. W. Williams (Eds), Butterworths, London, 1981, 57-76, Symposium Research and Development in Information Retrieval, Cambridge (UK), June 1980.

[van Rijsbergen 1979] C. J. van Rijsbergen: Information Retrieval. Butterworths, 2nd ed., London, 1979.

[Robertson 1995a] S. E. Robertson, S. Walker, M. M. Hancock-Beaulieu: Large Test Collection Experiments on an Operational, Interactive System: Okapi at TREC. Information Processing & Management, 31(3), 1995, 345-360.

[Robertson 1995b] S. E. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, M. Gatford: Okapi at TREC-3. Proceedings TREC'3, Gaithersburg (MD), April 1995, 109-126.

[Savoy 1995] J. Savoy: An Evaluation of Probabilistic Retrieval Models. Technical Report, Faculté de droit et des sciences économiques, Université de Neuchâtel, CR-I-95-04, June 1995, p. 29.

[Saracevic 1988] T. Saracevic, P. Kantor: A Study of Information Seeking and Retrieving. III. Searchers, Searches, Overlap. Journal of the American Society for Information Science, 39(3), 197-216.

[Shaw 1995] J. A. Shaw, E. A. Fox: Combination of multiple searches. Proceedings TREC'3, Gaithersburg (MD), April 1995, 105-108.

[Swanson 1977] D. R. Swanson: Information Retrieval as a Trial and Error Process. Library Quarterly, 47(2), 1977, 128-148.

[Tague-Sutcliffe 1995] J. Tague-Sutcliffe, J. Blustein: A Statistical Analysis of the TREC-3 Data. Proceedings TREC'3, Gaithersburg (MD), April 1995, 385-398.

[Tenopir 1985] C. Tenopir: Full Text Database Retrieval Performance. Online Review, 9(2), 1985, 149-164.

[Thompson 1990] P. Thompson: A Combination of Expert Opinion Approach to Probabilistic Information Retrieval. Information Processing & Management, 26(3), 1990, 371-394.

[Turtle 1991] H. Turtle, W. B. Croft: Evaluation of an Inference Network-Based Retrieval Model. ACM Transactions on Information Systems, 9(3), 1991, 187-222.

[Voorhees 1995] E. M. Voorhees, N. K. Gupta, B. Johnson-Laird: The Collection Fusion Problem. Proceedings TREC'3, Gaithersburg (MD), April 1995, 95-104.

[Voorhees 1995] E. M. Voorhees, N. K. Gupta, B. Johnson-Laird: Learning Collection Fusion Strategies. Proceedings SIGIR'95, Seattle (WA), July 1995, 172-179.

[Weiss 1991] S. M. Weiss, C. A. Kulikowski: Computer Systems That Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems. Morgan Kaufmann, San Mateo (CA), 1991.