From: Oddy et al. (eds), Information Retrieval Research. Butterworths, 1981

4

Probabilistic models of indexing and searching

S. E. Robertson, C. J. van Rijsbergen and M. F. Porter

4.1 Introduction

There has been a considerable amount of work in recent years on the probabilistic theory of document retrieval. The main sources for the present chapter are the work done in the UK on the searching end of the information storage and retrieval complex (*see*, for example, Robertson and Sparck Jones, 1976; van Rijsbergen, 1977; Harper and van Rijsbergen, 1978), and the work done in the USA on automatic indexing using within-document frequencies of terms (notably by Bookstein and Swanson, 1974, 1975; Harter, 1975a, b; Bookstein and Kraft, 1977). (There is a considerable body of related work by Salton, Yu and associates (*see*, for example, Salton, Wong and Yu, 1976), but the starting point for this is rather different, and it will therefore not be considered further here.)

The present study arises out of a project currently under way in Cambridge, involving the three authors. One of the main objects of the project is to bring together these two strands of work on indexing and searching. In particular, we hope to develop and test a model, within the framework of the probabilistic theory of document retrieval, which makes optimum use of within-document frequencies in searching. Progress towards this end, both theoretical and experimental, is described in this chapter.

As the work described here depends fairly heavily on previous work (by ourselves and others), we propose to spend a little time summarising the earlier theoretical and experimental results.

4.2 Theoretical background

4.2.1 Basic concepts

A fundamental part of the probabilistic theory of retrieval, as it has developed over the last ten years or so, is the Probability Ranking Principle (Robertson, 1977), which states that for optimum performance on a given query a document retrieval system should rank the documents in order of their probability of relevance (to the query or underlying need), according to the

information available to the system. This principle can be proved to hold provided that certain assumptions are made; it will be assumed here.

The core of the theory lies in the use of probability theory, together with assumptions about the statistical properties of the variables involved, to construct various retrieval functions which can be used to rank the documents in the appropriate order. The first step in such a construction is usually to invert the conditional probabilities by use of Bayes' theorem, as follows.

Assume that we have information about documents in the form of a variable, X, and assume that each document is assigned a value, x, of X. X may be regarded as a random variable associated with the population of documents. (X may take values in a multi-dimensional vector space, for example, each dimension being associated with an index term, and x is then a single vector describing the assignment of index terms to a particular document.) We have another variable of interest — namely, the relevance of a document to the query or underlying need. This variable is assumed to be dichotomous; it is denoted by A, with values 0 (for non-relevant) or 1 (for relevant).

We desire to rank the documents in order of their probabilities of relevance given the information provided by X — that is, in order of their values of

$$P(A=1|X=x)$$

It can be shown by Bayesian (van Rijsbergen, 1979) inversions that an identical ranking of the documents is produced by using the function

$$V(X=x)=\log\frac{f_X(x|A=1)}{f_X(x|A=0)}$$

where $f_X(x|A=a)$ is the density function of X in the population of relevant (or non-relevant) documents. If X is a discrete variable, then

$$f_X(x|A=a)=P(X=x|A=a)$$

that is, the probability of a relevant (or non-relevant) document having the value x of X. Since all the variables considered in this chapter are discrete, the form

$$V(X=x)=\log\frac{P(X=x|A=1)}{P(X=x|A=0)}$$

will be used.

One extremely useful property of these functions is that they are additive under independence. That is, if we have two variables X, Y, and wish to combine the information provided by them, then under assumptions about their independence in the relevance set and the non-relevance set, respectively, we have

$$\begin{aligned}V((X,\ Y)=(x,\ y))&=V(X=x \text{ and } Y=y)\\&=V(X=x)+V(Y=y)\end{aligned}$$

In other words, the function V forms a suitable basis for a sum-of-weights matching procedure, whereby each variable has associated with it a weight, and the total score of a document (by which it is ranked) is the sum of the component weights.

One final transformation is useful. It is often convenient, if a variable has a natural zero, to make sure that the weight associated with this value is zero. This can be done by constructing a retrieval function

$$W(X=x)=V(X=x)-V(X=0)$$

This function ranks the documents in the same order as V, and is also additive under independence.

To summarise: If the variable X is multi-dimensional with axes X_i, having vector values x with components x_i, then optimum retrieval performance (under the assumptions of the probability ranking principle) is ensured by ranking the documents in order of their associated values (scores) of

$$W(X=x)=\log\frac{P(X=x|A=1)P(X=0|A=0)}{P(X=x|A=0)P(X=0|A=1)} \tag{4.1}$$

Further, if independence between components of X is assumed, then

$$W(X=x)=\sum_i W(X_i=x_i)$$

where

$$W(X_i=x_i)=\log\frac{P(X_i=x_i|A=1)P(X_i=0|A=0)}{P(X_i=x_i|A=0)P(X_i=0|A=1)} \tag{4.2}$$

Equation (4.2) will form the basis for most of the retrieval functions discussed in this chapter.

4.2.2 Binary independence weights

We now concentrate on the case where each X_i corresponds to the assignment of index term i to a particular document, and where terms are either assigned or not, so that X_i has just two values. We define for a given query:

$p=P(t \text{ assigned}|A=1)$
$q=P(t \text{ assigned}|A=0)$

Then, from equation (4.2) and making the independence assumption, each term can be given a weight

$$W(t \text{ assigned})=\log\frac{p(1-q)}{q(1-p)} \tag{4.3}$$

This is the relevance weight used by Robertson and Sparck Jones (1976), and it will be referred to as the binary independence, or BI, weight. The question of how to obtain information about p and q, upon which any use of the formula must depend, is discussed in sub-sections 4.3.1–4.3.3.

4.2.3 Binary dependence weights

In 1977 van Rijsbergen proposed a weighting scheme which replaced the assumption of independence between index terms by one of partial dependence. Instead of just considering the absence or presence of individual terms independently, one *selects* certain pairs of terms and calculates a weight

for them jointly. If one assumes that term i depends significantly on j, then a document will receive a weight proportional to

$$\log \frac{P(X_i=x_i|X_j=x_j,\ A=1)}{P(X_i=x_i|X_j=x_j,\ A=0)} \tag{4.4}$$

It is crucial to the calculation of such a weight that one establish beforehand what are the significant dependencies between pairs of index terms. The theoretical model described in van Rijsbergen (1977, 1979) assumes that the important dependencies can be *selected* by constructing a spanning tree connecting the entire index term vocabulary. The spanning tree is chosen from a class of possible spanning trees in such a way that it optimises an objective function. Such a function can be a sum of similarities, where each similarity measures the extent of the dependence of a pair of terms connected by a link. The optimal spanning tree is the spanning tree for which the sum of similarities is maximised. Once the spanning tree has been established, only those pairs of terms directly connected are assumed to be significantly connected.

4.2.4 Harter's model

The discussion so far relates to recent work on probabilistic models, with the emphasis on the searching process. The work on indexing on which we propose to draw was developed independently, by Bookstein and Swanson (1974, 1975), Harter (1975a, b) and Bookstein and Kraft (1977). There are, however, several points of contact. This discussion will be based chiefly on Harter's two papers.

The simple version of Harter's model states that any content-bearing word will have within-document frequencies which fit a 2-Poisson distribution. That is, each word will have associated with it an 'elite' set of documents (sub-set of the collection); the within-document frequencies will follow a Poisson distribution within this elite set, and will follow a second Poisson distribution in the non-elite set formed by the rest of the documents, so the observed distribution on the entire collection is a mixture of the two. The elite set is either identified with, or assumed to be correlated with, the set of documents which would be judged relevant by a requester whose query was just this single word. (Single-term requests only are considered by Harter.)

Harter then develops a method for estimating, for any given word, the parameters of the two Poisson distributions. (The method is discussed further in sub-section 5.4, below.) He then uses a decision-theoretic argument to suggest criteria for deciding whether or not to assign any particular word as index term to any particular document. These criteria are based on a version of the probability ranking principle; this fact provides a point of departure for an attempt to unify the two lines of work.

Following is a condensed mathematical description of those features of Harter's model to which we shall be referring.

The variable K (with values k) is the number of occurrences of a given word in a document d. E represents the property of eliteness: $E=1$ means that the document belongs to the elite set; $E=0$ means that it does not. The 2-Poisson model requires three parameters for each word: the means of the two Poisson distributions, l (elite set) and m (non-elite set), and a mixing parameter h, defined by

$h = P(E=1)$

It is assumed that

$l > m$

(In effect, this is the definition of the elite set.)

The 2-Poisson model says that

$$P(K=k) = h\frac{\exp(-l)l^k}{k!} + (1-h)\frac{\exp(-m)m^k}{k!}$$

The indexing criteria depend on the parameter

$$P(E=1|K=k) = \frac{h\exp(-l)l^k}{h\exp(-l)l^k + (1-h)\exp(-m)m^k} \tag{4.5}$$

There are various versions of the indexing criteria, but the final one suggested by Harter uses the following definitions:

$$z = \frac{l-m}{(l+m)^{1/2}} \tag{4.6}$$

is a measure of the separation of the two descriptions;

$$b = P(E=1|K=k) + z \tag{4.7}$$

is a measure of the 'indexability', or the relative significance of the word in a document in which it occurs k times. The indexing criterion is then

index if and only if $b > 0$

The measure of indexability, b (Harter's beta), is arrived at by a somewhat *ad hoc* process, and, in fact, there is clearly a fault in the argument, since b is always greater than zero (unless $h=0$, which is a pathological case). However, Harter also suggests that b might be used as a weight; as we shall see below, this is one of the experiments we have tried.

4.3 Previous experimental results

4.3.1 Upper bounds

One use for a retrieval function such as the binary independence weight (4.3) which demands relevance information is to indicate an effective upper bound or optimum retrieval performance (Sparck Jones, 1975). This idea requires that we know in advance the relevant documents for each query (as is normally the case with test collections).

Suppose, then, that for a given query the entire collection of N documents contains R that are relevant; a particular term, t, is assigned to n documents, of which r are relevant. Then the obvious estimates for p and q are

$$\hat{p} = \frac{r}{R}; \qquad \hat{q} = \frac{n-r}{N-R}$$

From these and equation (4.3), we deduce that the binary independence weight associated with the term should be

$$W(t \text{ assigned}) = \log \frac{r(N-R-n+r)}{(R-r)(n-r)}$$

For reasons associated with estimation (Robertson and Sparck Jones, 1976) we more usually use a modified version:

$$W(t \text{ assigned}) = \log \frac{(r+0.5)(N-R-n+r+0.5)}{(R-r+0.5)(n-r+0.5)} \tag{4.8}$$

This solution to the estimation problem is not ideal (van Rijsbergen, Harper and Porter, in press) but will be used in this paper.

Experiments with equation (4.8) applied to the query terms only have consistently given high optimum performance (Robertson and Sparck Jones, 1976; Sparck Jones, 1979a, b, 1980). Higher performance still has been achieved with equation (4.4) applied to expanded queries (Harper and van Rijsbergen, 1978).

4.3.2 Relevance feedback

Realistically, it may be possible to exploit partial relevance information, in the form of relevance feedback, to improve a search statement for a subsequent search. Considering again the binary independence weights as an example, suppose that R now represents the *known* relevant documents (at a particular stage of the search), of which r are assigned term t. Then, again, we may estimate p by

$$\hat{p} = \frac{r}{R}$$

The appropriate estimate of q is not so obvious — we may take just those known to be non-relevant, or all those not known to be relevant. There is some evidence to suggest that the latter is preferable in terms of retrieval performance (Harper and van Rijsbergen, 1978), so the estimate of q takes the same form as before:

$$\hat{q} = \frac{n-r}{N-R}$$

and the binary independence weight is exactly as before (equation 4.8).

Two kinds of experiment are possible to evaluate relevance feedback strategies. The first is to perform an initial search on the document collection, select the top-ranking 10 or 20 documents to provide the feedback information, and then remove them from the collection for the subsequent search. This procedure is known as *residual ranking* (Ide, 1969; Harper and van Rijsbergen, 1978). The second is to divide the collection into two parts, obtain feedback information from one half and do the subsequent search on the other. This is known as the *half-collection* method (Robertson and Sparck Jones, 1976).

Experiments of both kinds with equation (4.8), again using query terms only, have once more consistently given good results. That is, it is possible to get substantial improvements in performance over non-feedback methods such as unweighted terms and collection-frequency weighting, even given very little feedback information (Sparck Jones, 1979a, b).

4.3.3 No relevance information

Croft and Harper (1979) have suggested using the binary independence weight without any relevance information. One might, in the absence of such information, give p a constant value (say 0.5), and estimate q by the overall frequency of the term — that is,

$$\hat{q}=\frac{n}{N}$$

These estimates, crude (and inconsistent) as they are, nevertheless do have some value. It turns out that when they are incorporated into equation (4.3), the resulting formula is closely related to the traditional collection-frequency weighting method, which has long been known to give performance improvements over unweighted terms (Sparck Jones, 1972).

In this chapter, we shall regard collection-frequency weighting by the formula

$$W(t \text{ assigned})=\log(N/n)$$

as an approximation to the special case of equation (4.3) where no relevance information is available.

4.3.4 Query expansion

The estimation problems encountered in implementing the independence weights become more severe for the dependence weights. If only 10 or 20 documents are used to provide the relevance feedback information, it is difficult to obtain reliable estimates for the dependence weights (equation 4.4). To overcome this difficulty, a hybrid model was proposed by Harper and van Rijsbergen (1978) in which dependence information was used to expand the query by selecting additional search terms from the spanning tree on the index term vocabulary. Once a query has been expanded, the weighting scheme can be that for the BI model (equation 4.3).

The feedback strategy for using dependence information between index terms is therefore now as follows. Construct a spanning tree on the term vocabulary using the distribution of co-occurrences throughout the entire document collection. For any given query, certain additional search terms are selected from the spanning tree by using the query terms as starting points. For example, one could include in the query the nearest neighbour connected in the tree to each query term. After the query has been expanded in this way, relevance weights can be calculated for each query term in the usual way.

Experimental results reported with this strategy have been conflicting. For some test collections, notably Cranfield 1400, query expansion has worked reasonably well, leading to significant improvements over non-expansion when performance is measured by residual ranking (Harper and van Rijsbergen, 1978). However, on the UKCIS data no significant improvement was observed (Harper, 1980). This leaves the whole question of the effectiveness of query expansion unresolved. It was always clear that any additional terms obtained by expansion would only be as good as the initial query terms. As yet no good heuristics for selecting query terms as candidates for expansion have been designed. Nor has the problem of which additional

search terms to include in the expanded query been investigated sufficiently. It would appear that a thorough investigation of the selection of both 'good' query terms and further 'good' search terms is needed.

4.3.5 Harter's experiments

No test of retrieval performance (like the experiments discussed above) has been done on the Harter indexing criterion. Instead, Harter compared the indexing generated by his criterion with indexing by a human indexer. Also, rather than use his criterion $b > 0$ to define an index set for each document, he simply ranked the words for each document in order of their b values, and observed the position of the human-assigned index terms in the ranked list. Thus, he used b as a weight (or something of that nature).

In the experiment, ranking by b value proved to be a reasonable prediction of whether the word had been assigned by the human indexer; in particular, it was a considerably better prediction than ranking words by k value — that is, simply by number of occurrences.

We do not propose to give a full account of Harter's experiment here, but some specific aspects should be mentioned. Although the theory deals with any kind of textual material, and therefore would have obvious application to collections of full texts of documents, Harter used a collection of abstracts only; we have done the same. Harter devised a method for estimating the parameters l, m, h of his model, involving the moments of the observed distribution, and including rules for dealing with exceptional cases; we have used a similar method.

Finally, Harter chose to do his analysis on words as they occur in texts, without any stemming or suffix-stripping operation. At this point we have departed from Harter's methods. The argument he gives is that different words from the same stem have different distributions. We dispute the argument on two grounds: first, whatever the distributions of the individual words, the stem itself might be supposed to have a 2-Poisson distribution and could therefore reasonably form the basis for a Harter-type analysis; second, the only final argument would be in terms of retrieval effectiveness. Given the evidence (in a somewhat different context) that stemming improves performance, we have decided to perform a stemming operation before applying the Harter model; but we suggest that some direct comparison would be desirable.

4.4 Main theoretical results

4.4.1 Relevance and eliteness

In tackling the problem of developing a Harter-type model for multi-term queries, we have to consider in some depth the status in the model of the property of relevance and also of the Harter idea of eliteness.

One possible approach would be to assume that there is some property equivalent to eliteness but relating to the complete set of terms used in a query. One would then need to postulate distributions for the various terms within the elite set and outside it, and perform a complete analysis of the raw occurrence and co-occurrence data for any combination of terms that occurred in a request. In general, this analysis would have to be performed *at*

the time of the request, and would have to be repeated if the system or the user were to expand the query. It seems, at least superficially, that this kind of process is unlikely to be a practical proposition. Nonetheless, an approach similar to this is taken by Bookstein and Kraft (1977): they suggest ways of selecting likely combinations of terms beforehand.

An alternative approach is to associate elite sets with the individual terms and assume a more complex model relating the various elite sets to the one relevance set for the query. This enables the analysis to be performed as an indexing operation, prior to the processing of requests. This is the approach adopted here.

4.4.2 Relations assumed in the model

We assume, then, that each term has associated with it an elite set and that the distributions of numbers of occurrences of the term in the two sets (elite and non-elite) are different (in particular, we assume that both are Poisson). We assume further that the elite sets for the query terms (and perhaps others) are correlated with relevance to the query, in a manner to be specified below.

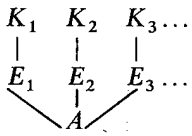
We wish to assume that some of the possible relationships between the variables of concern to us do *not* exist; that is, we wish to assume statistical independence in some cases. There are several different (but equivalent) ways of formulating an assumption of statistical independence, and it is worth indicating the differences and equivalences. Suppose that a, b, c are three events; then the following statements are equivalent:

$$P(a, b|c) = P(a|c)P(b|c)$$
$$P(a|b, c) = P(a|c)$$
$$P(b|a, c) = P(b|c)$$

All the statements say (in words) that a and b are independent, given c; or, equivalently, that the only relation between a and b is that implied by c. Diagrammatically, one might represent the situation under such an assumption thus:

That is, there is assumed to be no *direct* relation between a and b.

Returning to the 2-Poisson model, we can represent our independence assumptions thus: Suppose that there is a query with relevance property A, with terms $t_1, t_2, t_3 \ldots$ each with eliteness E_i and number of occurrences K_i; then we assume:



The various assumptions embedded in this model are:

(1) The number of occurrences of a term depends only on the property of eliteness for that term, not on eliteness for any other term or on relevance.

This is precisely equivalent to assumptions made by previous authors with this model.

(2) Eliteness for a given term depends only on relevance, not on eliteness for any other term. This is equivalent to the independence assumptions that were used in the derivation of equation (4.2).

Like all independence assumptions, these must be regarded as simplifications which we make in order to render the mathematical model tractable and to which it may be profitable to return with a view to reducing or eliminating the implied constraints. In fact, Bookstein and Kraft (1977) consider a rather more general model which (in the absence of variable A) assumes that E_i and E_j may be related.

As mentioned above, these assumptions enable us to estimate the 2-Poisson parameters separately for each term, as an indexing operation, and then to use the results of this analysis in searching.

4.4.3 2-Poisson independence weights

As with the binary independence model, we need two parameters to describe the relation between eliteness for a term and relevance. These are as follows (for a given term):

$$p' = P(E=1|A=1)$$
$$q' = P(E=1|A=0)$$

Now we are in a position to calculate the components of a weight to be assigned to any specific number of occurrences (value of K) for this term, according to formula (4.2). We have:

$$\begin{aligned} P(K=k|A=1) &= P(K=k|E=1)P(E=1|A=1) \\ &\quad + P(K=k|E=0)P(E=0|A=1) \\ &= \frac{1}{k!}(p' \exp(-l)l^k + (1-p')\exp(-m)m^k) \end{aligned}$$

and similarly for $A=0$ and q' (the first step in this derivation makes use of independence assumption 1). Hence, the full expression for $W(K=k)$, according to formula (4.2), becomes:

$$W(K=k) = \log \frac{P_k Q_0}{P_0 Q_k} \tag{4.9}$$

where

$$P_k = p' \exp(-l)l^k + (1-p')\exp(-m)m^k$$
$$Q_k = q' \exp(-l)l^k + (1-q')\exp(-m)m^k$$
$$P_0 = p' \exp(-l) + (1-p')\exp(-m)$$
$$Q_0 = q' \exp(-l) + (1-q')\exp(-m)$$

If we make independence assumption (2), then this weight (the 2-Poisson independence or TPI weight) gives optimal retrieval performance according to probabilistic retrieval theory.

Expression (4.9) is a pretty alarming one, not so much because of its size and inherent complexity (which is no problem to a computer), as because of the

number of different bits of information it requires. In particular, it needs the estimates of l, m from the Harter-type analysis, query specific estimates of p' and q' (which, like p and q, might be taken from relevance feedback data) and values of K for each document–term pair. However, there is a sufficient theoretical justification to attempt some experiments (at least) on the formula, even if we may subsequently look for some simplification of it for practical purposes.

4.4.4 Estimation of TPI

As with BI, we suppose first of all that we have complete relevance information: that is, we know the set of relevant documents (which have $A=1$). Unfortunately, we do *not* know the elite set for a term exactly: our knowledge of it is probabilistic, based on term occurrence data. So the estimation of p' and q' is not quite as obvious as that of p and q. The following argument, however, provides us with plausible estimates.

For a given term we can calculate the probability that any specific document belongs to the elite set, on the basis of the number of occurrences of the term in the document:

$$u_k = P(E=1|K=k)$$

for which an expression has already been given (equation 4.5).

Given the relevant documents, we wish to estimate

$$p' = P(E=1|A=1)$$

and an appropriate estimate would be

$$\hat{p}' = \frac{\text{number of relevant documents in the elite set}}{\text{total number of relevant documents}}$$

We do not know the numerator, but we can obtain an *expected* value for it: the sum, over the relevant documents, of the probability of each document belonging to the elite set. Using this value, we can write

$$\hat{p}' = \frac{\sum A u_k}{\sum A}$$

where in each case the sum is over all documents. Since u_k is the same for all documents with the same K value, we can cumulate by K value:

$$\hat{p}' = \sum_k f_k u_k \tag{4.10}$$

where f_k is the proportion of relevant documents having k occurrences of the term. Similarly,

$$\hat{q}' = \sum_k f_k u_k \tag{4.11}$$

where f_k is now the proportion of non-relevant documents having $K=k$.

Again following the BI example, we may reasonably use estimates of the same form if we have only partial relevance information. Then f_k in equation (4.10) would be the proportion of *known* relevant documents having $K=k$, and

in equation (4.11) it would refer to the remaining documents (the complement set).

It is also possible, in the absence of any relevance information, to make plausible guesses for p' and q', following the example of Croft and Harper's (1979) guesses for p and q. Some such guesses are discussed below (in subsection 4.5.9).

4.4.5 Mixability of independence weights

One further comment should be made here about the general weighting scheme discussed above. The breaking down of formula (4.1) into its components (4.2) under independence assumptions does *not* depend on the various components being similar in nature. One might, for example, have some components (say words in text) which have 2-Poisson characteristics, and thus should be weighted according to equation (4.9), and others (say assigned index terms or authors or citations) which are essentially binary, and thus should be weighted according to equation (4.3). The two sets of weights would then (according to the theory) be strictly compatible, and could therefore be used in combination in weighted retrieval.

No experiments with such mixings are reported here, but the theoretical result is a powerful and potentially important one.

4.5 Experiments

4.5.1 Overview of the experiments

In this section we present the results of a series of experiments on one test collection. The purpose of these experiments was to test some of the theoretical ideas discussed above. In particular, we have:

(1) Repeated some earlier experiments, using binary independence weights on unexpanded and expanded queries.
(2) Performed a Harter-type analysis of distributional characteristics of terms.
(3) Tested some simple weighting schemes based on the Harter model (query terms only).
(4) Tested the full 2-Poisson independence weights in various ways (query terms only).
(5) Tested the use of some Harter-type indexing criteria.

Descriptions of the test collection used and the experimental method follow.

4.5.2 The NPL data

The NPL test collection consists of the titles and abstracts of about 11 000 documents, together with a set of queries and corresponding relevance assessments. The material was originally prepared by Vaswani and Cameron at the National Physical Laboratory, and this is described on pages 9–13 of their report (1970). For the purpose of experimentation, we have represented each document by a vector of the terms which index it, and the terms are derived from the title and abstract for the document by discarding the

TABLE 4.1.

	Documents		*Queries*		*Relevance assessments*	
		inv.		inv.		inv.
Number	11429	7491	93	337	93	1735
Maximum length	105	2511	13	19	84	6
Minimum length	1	1	2	1	1	1
Total length	228087	228087	664	664	2083	2083
Average length	19·96	30·45	7·14	1·97	22·40	1·20

common words ('and', 'are', 'by', etc.) and applying a stemming algorithm (see Porter, 1980) to the remainder. A document d is therefore indexed by a list of terms $t_1, t_2, \ldots$, where each t_i corresponds to a word stem derived from the original text representative for the document. t_i will have a document frequency k_i, the number of times the stem of t_i occurs in the document representative. n_i will denote the number of documents in which t_i occurs, and we will usually drop the suffixes, so that k and n are frequency parameters of term t.

The queries are set up as vectors of terms in an analogous way, and the relevance assessments consist of vectors of document numbers. *Table 4.1* provides a brief summary of the NPL data. Thus, the third column shows that there are 93 queries, with an average length of 7.14 terms (total length = 664), ranging from 2 to 13 terms. The columns headed 'inv.' show the statistics for the inverted structures. Thus, the inverse of the term vectors of documents is a set of vectors containing document numbers, which give the documents in which the various terms occur, and the second column shows that there are 7491 terms in the collection, and each term occurs in an average of 30.45 documents, the most frequent term (highest n value) occurring in 2511 documents.

A number of changes were made to the original text of the collection in the form in which we received it. About 1900 edits were applied to the text to remove some of the gross typing errors. This task had been beyond the resources of the original investigators. One hundred and forty-two documents were removed which had been inexplicably duplicated in the collection, and the document numbers were reduced accordingly. Finally, the queries with empty relevance assessments were discarded.

4.5.3 Appropriateness of the NPL test collection

The NPL test collection is an unusually large one. This was important in testing the Harter model, since a substantial amount of term frequency information was required in order to be able to get good estimates of the parameters l, m and h for each term. Three points, however, must be made:

(1) The term-frequency information is derived from document abstracts, not from the entire document text. Following Harter, we felt that this was acceptable, since, like the document text, the abstract is a piece of continuous prose containing words which will fall into two classes —

those appropriate and those not appropriate for indexing the document. Furthermore, we were naturally interested in the applicability of the model to the typical IR environment, where the document representative will contain little more than the document abstract.

(2) The removal of the common words ('and', 'are', 'by', etc.) could be left to the Harter model. The list of common words, however, is short (just 251) and the words in it are very neutral in meaning, and it was felt to be unexceptionable to remove them at an earlier stage.

(3) The Harter model is being applied here to word-stems rather than words. Harter's argument for using words is discussed above.

4.5.4 Estimating the Harter parameters

In deriving l, m and h for each term, Harter's estimation method was used, whereby l and m are the roots of

$$al^2+bl+c=0$$

where

$$a=M^2-L, \quad b=K-LM, \quad C=L^2-MK$$

and

$$M=R_1, \quad L=R_2-R_1, \quad K=R_3+2R_1-3R_2$$

R_1, R_2, R_3 being the first three sample moments of the distribution. h is given by

$$h=\frac{M-m}{l-m}$$

Various degenerate cases can arise, which are dealt with as follows: if $b^2-4ac<=0$, set $l=M$ and $m=0$; if $m<0$, set $l=L/M$ and $m=0$; and then, unless $0<h<=1$, set $l=M$ and $m=0$, and recompute h.

In fact, for the majority of these terms, one or other of these degenerate cases will apply. This is because, for the low-frequency terms, there are insufficient data to separate out the two Poisson distributions, and the Zipfian distribution of the terms guarantees that the majority will be of low frequency. But this does not necessarily matter, since the 'important' terms in a collection (those that get used in queries) are usually of high to middle frequency.

As an illustration of the use of Harter's model with the NPL data, *Table 4.2* gives a list of 40 nearly equifrequent terms taken as a continuous batch from the term-frequency ranking, and arranged by decreasing z, Harter's measure of term effectiveness (equation 4.6). Each term is given by a representative word corresponding to the stem.

4.5.5 Presenting the experiments

In each of the experiments in which the queries are matched against the documents, recall and precision values are obtained using the standard recall

TABLE 4.2.

Term	*z value*	*Term*	*z value*
1 RESPONSE	1·41	21 INCREASE	0·72
2 MAXIMUM	1·20	22 PROBLEM	0·70
3 CHARGE	1·18	23 MICROWAVE	0·69
4 ELEMENT	1·09	24 LINEAR	0·64
5 COUPLE	1·07	25 PROPAGATE	0·57
6 IMPEDANCE	1·07	26 NEW	0·57
7 RECORD	1·03	27 SIMPLE	0·55
8 SOURCE	0·99	28 TECHNIQUE	0·52
9 PLASMA	0·99	29 VARIOUS	0·46
10 AMPLITUDE	0·94	30 DEPEND	0·45
11 VALVE	0·90	31 PRODUCE	0·42
12 CONSTANT	0·89	32 EXPRESS	0·40
13 COMPONENT	0·84	33 DUE	0·40
14 DIRECT	0·81	34 INCLUDE	0·37
15 NUMBER	0·78	35 POSSIBLE	0·35
16 APPROXIMATE	0·78	36 ELECTROMAGNET	0·34
17 FORMULA	0·75	37 GIVE	0·28
18 CHANGE	0·74	38 ACCOUNT	0·26
19 TERM	0·73	39 REPORT	0·23
20 PERIOD	0·72	40 SEE	0·18

cutoff method (van Rijsbergen, 1979). A result can be presented in the form of a simple table as follows:

(1) Co-ordination level match

Recall	*Precision*
0	59.91
10	49.07
20	37.62
30	30.59
40	24.94
50	20.36
60	13.22
70	10.61
80	7.23
90	4.72
100	2.23

This shows the percentage precision at the percentage recall values 0, 10 ... 100 using a co-ordination level match on the NPL collection. The label (1) will identify the experiment. To abbreviate this table, the recall column will be omitted; the precision will be given to the nearest per cent; the recall 0 value, which is untrustworthy, and the recall 100 value, which is relatively uninformative, will be omitted; and the whole will be written on one line as:

49 38 31 25 20 13 11 7 5 [1] C

where C indicates 'co-ordination level match', and the experiments are

identified by a number in square brackets possibly followed by a letter — for example, [5x]. When an experiment is being presented merely for comparison, it is prefixed with 'cf.', thus

54 45 37 31 24 18 15 10 6 [2] log(N/n)
cf. 49 38 31 25 20 13 11 7 5 [1] C

In [2] each term is given a simple form of collection frequency weight, namely log(N/n), where N is the collection size.

4.5.6 Collection frequency experiments

A number of experiments were performed using the term weight log(N/n), so as to provide a basis for comparison with other retrieval methods. The document collection was split into two halves, the even-numbered documents constituting one half (the E half), and the odd-numbered documents constituting the other half (the O half). In experiments [2a] and [2p] the O half only was used, and in experiments [2b] and [2q] the E half only. In experiments [2p] and [2q], however, the values of N and n for each term weight were derived from the whole collection, while in experiments [2a] and [2b] they were derived from the O and E halves, respectively. The results were as follows:

57 48 39 33 28 21 17 12 8 [2p] log(N/n) to O
57 49 39 33 28 21 17 12 8 [2a] log(N/n) O to O
cf. 54 45 37 31 24 18 15 10 6 [2] log(N/n)

56 48 38 31 26 23 17 14 10 [2q] log(N/n) to E
57 48 39 30 26 22 17 14 10 [2b] log(N/n) E to E
cf. 54 45 37 31 24 18 15 10 6 [2] log(N/n)

'E to E' means that the parameter estimates were derived from the E half and that the retrieval run was done on the E half, 'to E' means that they were derived from the whole collection and applied to the E half, and so on. Although performance on the O and E halves differ from each other and also from performance on the whole collection, it will be seen that [2a] nearly equals [2p] and that [2b] nearly equals [2q]. This type of equality will be assumed in the sequel, where, in experiments on the half collections involving the Harter model, the Harter parameters are, in fact, estimated from the whole collection.

4.5.7 Binary independence weighting experiments

Seven experiments were performed on the BI weight using formula (4.8). In [3] the BI weights were derived from the entire relevance set. In [3a] the BI weights were derived from the relevance set in the O half and then the retrieval run was performed on the O half. In [3d] the BI weights were again derived from the relevance set in the O half, but then the retrieval run was performed on the E half. [3b] and [3c] correspond to [3a] and [3d] with O and E interchanged. In [3e] the BI weights were derived from the relevant documents occurring in the top 20 rank positions in a co-ordination level match on the E half. The retrieval run was then performed on the O half. [3f] is [3e] with O and E interchanged.

Experiments [3], [3a] and [3b] can be thought of as providing upper

bounds for retrieval performance using BI weighting. [3e] and [3f] simulate a retrieval situation in which the user discovers relevant documents among the D highest ranking documents by co-ordination level match (here $D=20$), and this relevance information is used by the system in a further retrieval run. [3c] and [3d] provide upper bounds on the performance of [3e] and [3f] as D is allowed to increase to infinity. The results are as follows:

69 59 51 44 37 28 22 16 10 [3] BI

70 63 55 47 42 34 27 18 12 [3a] BI O to O
65 56 46 38 34 26 21 15 9 [3c] BI E to O
59 50 40 33 29 22 18 12 8 [3e] BI E/20 to O

73 66 57 49 42 35 26 23 16 [3b] BI E to E
62 56 49 39 34 29 21 17 12 [3d] BI O to E
60 52 43 35 30 25 17 14 10 [3f] BI O/20 to E

4.5.8 Simple weighting from the Harter model

Several simple weighting formulae are suggested (in a somewhat *ad hoc* fashion) by the Harter model (simple in the sense that they use the distributional characteristics of terms, but no individual within-document frequencies).

The parameter h in the Harter model plays a similar role to collection frequency (n/N) in ordinary binary indexing. Since collection frequency weighting is known to be of value, this suggests trying $-\log h$ as a weight:

55 45 37 30 25 19 14 10 6 [4] $-\log h$
cf. 54 45 37 31 24 18 15 10 6 [2] $\log(N/n)$

As predicted, $-\log h$ provides a weight comparable in performance to collection frequency.

Now Harter's z measure:

cf. 54 45 37 31 24 18 15 10 6 [2] $\log(N/n)$
54 43 35 29 23 16 12 9 6 [5] z
cf. 49 38 31 25 20 13 11 7 5 [1] C

There are good theoretical reasons (Croft and Harper, 1979) to suggest that a successful term weight will have a collection frequency type of distribution. Harter's z weight does not have this, and the placing of [5] between [1] and [2] indicates that z gives a useful discrimination to equifrequent terms but does not scale correctly over the entire range of terms with different frequencies. An attempt can be made to force a collection frequency type of distribution by multiplying the z weight by $\log(N/n)$. The result is:

cf. 69 59 51 44 37 28 22 16 10 [3] BI
60 50 41 33 26 20 15 11 7 [6] $z\log(N/n)$
cf. 54 45 37 31 24 18 15 10 6 [2] $\log(N/n)$

This is an attractive result, giving, at the low-recall end, a performance mid-way between the collection frequency weight and the BI upper bound. In fact, the weight $z\log(N/n)$ operating on the half collections will outperform the relevance feedback experiments [3e] and [3f]:

61 52 39 34 29 22 18 13 8 [6a] $z\log(N/n)$ to O
cf. 59 50 40 33 29 22 18 12 8 [3e] BI E/20 to O

63 54 44 37 32 27 20 15 11 [6b] $z\log(N/n)$ to E
cf. 60 52 43 35 30 25 17 14 10 [3f] BI O/20 to E

However, it is unfortunate that we have not been able to derive a weight of this kind directly from the Harter theory.

A more complex (but still *ad hoc*) weight is the b weight (Harter's beta) defined above (equation 4.7). This does use the individual within-document frequencies, K, and performs as follows:

52 44 35 30 23 16 12 9 6 [7] b
cf. 54 43 35 29 23 16 12 9 6 [5] z

60 51 41 34 28 20 15 11 7 [8] $b\log(N/n)$
cf. 60 50 41 33 26 20 15 11 7 [6] $z\log(N/n)$

It can be seen that b and z perform in about the same way. This means that we are not making any effective use here of the within-document information about each term.

4.5.9 2-Poisson independence weights

In the formulae of equations (4.10) and (4.11) p' and q' are derived from a certain set of relevant documents and the complement of that set. Calling either set S, p' and q' are estimated by

$$\sum_k f_k u_k$$

where f_k is the proportion of documents in S having k occurrences of term t, and u_k is $P(E=1|K=k)$. The summation here does not involve many terms, since f_k quickly becomes 0 as k increases.

The experiments conducted with the TPI weighting scheme (equation 4.9) are entirely analogous to experiments [3] to [3f], and are as follows:

71 60 49 42 35 27 20 15 10 [9] TPI

74 67 56 48 43 36 28 20 15 [9a] TPI O to O
61 53 42 36 32 25 19 13 8 [9c] TPI E to O
57 48 38 32 28 21 16 10 6 [9e] TPI E/20 to O

74 64 55 49 42 35 26 22 17 [9b] TPI E to E
61 51 44 37 32 27 21 17 12 [9d] TPI O to E
61 53 43 35 29 24 18 14 10 [9f] TPI O/20 to E

A comparison with [3] to [3f] shows that usually the TPI in its present form performs about the same as (perhaps slightly worse than) the BI model.

A simpler approach to the TPI weight is to make plausible guesses for the values of p' and q'. The guesses $p'=1$ and $q'=0$ have the merit of simplifying the formula to $k\log(l/m)$, although as guesses they are not too plausible, since they involve the assumption that for each query term the elite set is the relevant set. Despite this, $k\log(l/m)$ does even worse than we expected. On the other hand, it was found empirically that $\log(l/m)$ as a weight (another simple weighting scheme) does quite well:

32 25 19 16 13 10 9 7 5 [10] $k\log(l/m)$
56 46 37 28 23 18 14 10 6 [11] $\log(l/m)$

The natural guess for q' is h. With p' then ranging from 0 to 1, various performance figures were obtained, which outperformed collection frequency weighting for $p' >= 0.3$, and which reached a peak at about $p' = 0.5$. The $p' = 0.5$ estimate performed as follows:

57 48 40 32 26 21 16 11 8 [12] TPI $p' = 0.5$ $q' = h$
cf. 54 45 37 31 24 18 15 10 6 [2] $\log(N/n)$

Comparing experiment [12] with the best simple weight $z\log(N/n)$ (neither using relevance information), we find similar overall levels of performance but differently shaped curves:

cf. 60 50 41 33 26 20 15 11 7 [6] $z\log(N/n)$
cf. 57 48 40 32 26 21 16 11 8 [12] TPI $p' = 0.5$ $q' = h$

4.5.10 Harter indexing experiments

In the main document collection, and, in particular, for all the experiments involving binary indexing, each document is indexed by every term within it. Instead of using Harter-model ideas to weight the terms, we might follow Harter's original suggestion of applying indexing criteria to decide whether or not to apply a term under given conditions. This could have the effect of reducing the collection — that is, of discarding some term assignments previously made.

As has been noted above, Harter's final indexing criterion contains an obvious mistake, and so cannot be used directly. His other criteria involve assessments of cost to the user, and so again are not accessible to us in this case. Instead, we tried two very simple criteria:

(1) index if and only if

$$P(E=1|K=k) > P(E=0|K=k)$$

or equivalently

$$P(E=1|K=k) > 0.5$$

(2) index if and only if

$$P(E=1|K=k) > P(E=1)$$

or equivalently

$$P(E=1|K=k) > h$$

and some minor variants on these criteria.

The indexing strategy could then be evaluated by comparing the performances of the same retrieval method used with the main and the reduced document collections. With these experiments the results have been entirely negative: removal of terms from the main file degraded performance, and the more terms removed, the greater the degradation became.

4.5.11 Query expansion experiments

Our results on query expansion using the NPL data are disappointing. We have not been able to achieve any significant improvements over non-expansion. We have repeated experiments done previously in which the query was expanded and the resulting set of search terms then weighted by BI. Once again the results have been conflicting. On the one hand, when the feedback experiment is evaluated by residual ranking the performance of the expansion does not differ significantly from that for non-expansion. On the other hand, using the half-collection technique, we found that query expansion degraded performance. The reason for this remains unclear and needs further investigation.

4.6 Discussion

The theory of probabilistic retrieval can be applied to various bits of information about the documents. The simplest idea is to apply it to query terms only, and binary indexing (terms present or absent). Beyond this, one might consider various additional bits of information — in particular, index terms other than query terms, or within-document frequencies of terms. For the latter we can imagine three different ways of using within-document frequency information:

(1) Locally only — that is, to use the occurrences of terms in a particular document to aid in the decision whether to retrieve that document or not.
(2) Globally — that is, to use the statistical properties of occurrence distribution of terms as a guide to the use of those terms in retrieval.
(3) Globally and locally — that is, to combine general statistical characteristics of terms with document-specific information.

The 2-Poisson model on which much of the work in this chapter is based does not have anything to say about possibility (1). It does suggest (in a rather *ad hoc* manner) some possibilities for (2), but principally it leads to (3). Again there are two possibilities for (3) — namely that we use the information for selective term assignment (as suggested by Harter) or for index term weighting. The major theoretical contribution of this chapter is the development of an appropriate weighting scheme. Experimentally, we have tested some of the ideas on one test collection. All conclusions must be tentative at best until other collections have been tried. However, the principal conclusions to be drawn from this series of experiments are as follows:

(1) There is benefit to be gained from using within-document frequency on a global basis.
(2) The local and global use of within-document frequencies appears to have potential, but does not (in the form proposed here) match simpler methods.
(3) Our experiments confirm earlier ones on the difficulty of gaining substantial improvements by expanding the query.

A possible reason for the slightly disappointing performance of the 2-Poisson independence weighting scheme lies in the problem of estimation. This area is

known to cause difficulties with binary independence weights, and there are many more parameters to estimate with TPI.

Apart from repeating some of the experiments on different collections, we might therefore suggest two further lines of work. The first is to delve more deeply into the estimation problems, and the second is to look for a better theoretical explanation of the performance of the global-only methods. For the latter it seems likely that these methods work well because they are related in some way to TPI (as collection frequency is related to BI).

Acknowledgements

The authors are grateful to the British Library Research and Development Department for the funds which supported this work, to the National Physical Laboratory for the use of the test data and to the University of Cambridge Computer Laboratory for computing resources and a congenial research environment.

References

BOOKSTEIN, A. and KRAFT, D. (1977). 'Operations research applied to document indexing and retrieval decisions', *Journal of the ACM*, **24**, No. 3, 418–427

BOOKSTEIN, A. and SWANSON, D. R. (1974). 'Probabilistic models for automatic indexing', *Journal of the ASIS*, **25**, No. 5, 312–319

BOOKSTEIN, A. and SWANSON, D. R. (1975). 'A decision theoretic foundation for indexing', *Journal of the ASIS*, **26**, 45–50

CROFT, W. B. and HARPER, D. J. (1979). 'Using probabilistic models of document retrieval without relevance information', *Journal of Documentation*, **35**, 285–295

HARPER, D. J. (1980). *Relevance Feedback in Document Retrieval*, Ph.D. Thesis, University of Cambridge

HARPER, D. J. and VAN RIJSBERGEN, C. J. (1978). 'An evaluation of feedback in document retrieval using co-occurrence data', *Journal of Documentation*, **34**, 189–216

HARTER, S. P. (1975a). 'A probabilistic approach to automatic keyword indexing. Part I: On the distribution of specialty words in a technical literature', *Journal of the ASIS*, **26**, 197–206

HARTER, S. P. (1975b). 'A probabilistic approach to automatic keyword indexing. Part II: An algorithm for probabilistic indexing', *Journal of the ASIS*, **26**, 280–289

IDE, E. (1969). *Relevance Feedback in an Automatic Document Retrieval System*, M.Sc. Thesis, Report ISR-15 to the National Science Foundation, Department of Computer Science, Cornell University, Ithaca, N.Y.

PORTER, M. F. (1980). 'An algorithm for suffix stripping', *Program*, **14**, 130–137

ROBERTSON, S. E. (1977). 'The probability ranking principle in IR', *Journal of Documentation*, **33**, 294–304

ROBERTSON, S. E. and SPARCK JONES, K. (1976). 'Relevance weighting of search terms', *Journal of the ASIS*, **27**, 129–146

SALTON, G., WONG, A. and YU, C. T. (1976). 'Automatic indexing using term discrimination and term precision measurement', *Information Processing and Management*, **12**, 43–51

SPARCK JONES, K. (1972). 'A statistical interpretation of term specificity and its application in retrieval', *Journal of Documentation*, **28**, 11–21

SPARCK JONES, K. (1975). 'A performance yardstick for test collections', *Journal of Documentation*, **31**, 266–272

SPARCK JONES, K. (1979a). 'Experiments in relevance weighting of search terms', *Information Processing and Management*, **15**, 133–144

SPARCK JONES, K. (1979b). 'Search term relevance weighting given little relevance information', *Journal of Documentation*, **35**, 30–48

SPARCK JONES, K. (1980). 'Search term weighting: some recent results', *Journal of Information Science*, **1**, 325–332

VAN RIJSBERGEN, C. J. (1977). 'A theoretical basis for the use of cooccurrence data in information retrieval', *Journal of Documentation*, **33**, 106–119

VAN RIJSBERGEN, C. J. (1979). *Information Retrieval* (2nd edn), Butterworths, London

VAN RIJSBERGEN, C. J., HARPER, D. J. and PORTER, M. F. (in press). 'The selection of good search terms', *Information Processing and Management*

VASWANI, P. K. T. and CAMERON, J. B. (1970). *The National Physical Laboratory Experiments in Statistical Word Associations and Their Use in Document Indexing and Retrieval*, National Physical Laboratory, Teddington